الجمهورية الجزائرية الديمقراطية الشعبية
وزارة التعليم العالي و البحث العلمي
جامعة ابن خلدون – تيارت –
كلية العلوم الاقتصادية و التجارية و علوم التسيير
قسم العلوم الاقتصادية

مطبوعة بعنوان

محاضرات في مقياس الاعلام الآلي 2

موجهة لطلبة السنة الثانية التخصصات التالية:

- العلوم الاقتصادية
- العلوم التجارية
- علوم التسيير
- علوم مالية و محاسبة

من اعداد الدكتور: ستي حميد

السنة الجامعية: 2018 – 2019

الفهـــرس

المحتــوى	الصفحة
مقدمــة	أ
الفصـــل الأول: مدخــل إلــى لغـــة البرمجـــة PASCAL	**01**
01.01 – تمهيد	02
02.01 – أول برنامج بلغة الـ PASCAL	02
03.01 – بنية (هيكل) برنامج بلغة الـ **PASCAL**	03
01.03.01 – L'en-tête	03
02.03.01 – جزء التصريح	03
03.03.01 – جزء التعليمات	05
04.01 – Les Identificateurs	08
05.01 – Les Identificateurs Prédéfinies	10
06.01 – Les Mots Clés Réservés	10
07.01 – الشكل الحر Le Format Libre	11
08.01 – التعاليق Les Commentaires	11
تمارين الفصل	13
الفصـــل الثانـــي: مفهـــوم المعطيـــات: المتغيـــرات و الثـــوابت	**16**
01.02 – تمهيد	17
02.02 – المعطيات	17
03.02 – أنواع المعطيات	18
01.03.02 – المعطيات الصحيحة	19
02.03.02 – المعطيات الحقيقية	20
03.03.02 – المعطيات سلسلة أحرف	20
04.03.02 – المعطيات من النوع حرف	21
04.02 – المتغيرات	21
05.02 – أنواع المتغيرات	23
01.05.02 – المتغيرات الصحيحة	23
02.05.02 – المتغيرات الحقيقية	23
03.05.02 – المتغيرات سلسلة أحرف	24

04.05.02 – المتغيرات من النوع حرف	24
05.05.02 – المتغيرات البولية Booléennes	25
06.02 – العمليات على المتغيرات (المعطيات)	25
01.06.02 – العمليات على المتغيرات الصحيحة	25
02.06.02 – العمليات على المتغيرات الحقيقية	27
03.06.02 – العمليات على المتغيرات سلسلة أحرف	28
04.06.02 – العمليات على المتغيرات من النوع حرف	28
05.06.02 – العمليات على المتغيرات من النوع Booléenne	29
07.02 – الثوابت	29
تمارين الفصل	30
الفصـــل الثالـــث: التصريـــح بالمعطيـــات	**33**
01.03 – تمهيد	34
02.03 – التصريح بالمتغيرات	34
01.02.03 – التصريح بالمتغيرات الصحيحة	35
02.02.03 – التصريح بالمتغيرات الحقيقية	38
03.02.03 – التصريح بالمتغيرات من النوع سلسلة أحرف	41
04.02.03 – التصريح بالمتغيرات من النوع حرف.	44
03.03 – التصريح بالثوابت	47
01.03.03 – التصريح بالثوابت الصحيحة	48
02.03.03 – التصريح بالثوابت الحقيقية	50
03.03.03 – التصريح بالثوابت من النوع سلسلة أحرف	53
04.03.03 – التصريح بالثوابت من النوع حرف	55
تمارين الفصل	59
الفصـــل الرابـــع: إدخـــال المعطيـــات	**77**
01.04 – تمهيد	78
02.04 – إدخال المعطيات عن طريق البرنامج (التوجيه المباشر)	78
01.02.04 – إدخال المعطيات الصحيحة	79

02.02.04 – إدخال المعطيات الحقيقية	82
03.02.04 – إدخال المعطيات سلسلة أحرف	86
04.02.04 – إدخال المعطيات من النوع حرف	88
03.04 – إدخال المعطيات عن طريق لوحة المفاتيح	91
01.03.04 – إدخال المعطيات الصحيحة	91
02.03.04 – إدخال المعطيات الحقيقية	98
03.03.04 – إدخال المعطيات سلسلة أحرف	104
04.03.04 – إدخال المعطيات من النوع حرف	107
04.04 – إدخال المعطيات مع التعليق	110
تمارين الفصل	111
الفصـــل الخـامس: معالجـة المعطيـات العدديـة	**132**
01.05 – تمهيد	133
02.05 – معالجة المعطيات العددية	133
03.05 – العمليات الحسابية و العلامات الحسابية	133
01.03.05 – عملية الجمع و العلامة + ...	133
02.03.05 – عملية الطرح و العلامة –	141
03.03.05 – عملية الضرب و العلامة *	146
04.03.05 – عملية القسمة الحقيقية و العلامة /	150
05.03.05 – عملية القسمة الصحيحة و العلامة DIV	153
06.03.05 – باقي عملية القسمة و العلامة MOD	155
07.03.05 – عملية الأس و العلامة **	156
04.05 – الدوال على المعطيات العددية	158
01.04.05 – الدالة ( )SQRT	159
02.04.05 – الدالة ( )SQR	163
03.04.05 – الدالة ( )ABS	168
04.04.05 – الدالة ( )LN	172
05.04.05 – الدالة ( )INT	177

06.04.05 – الدالة ( ) TRUNC	180
07.04.05 – الدالة ( ) ROUND	181
08.04.05 – الدالة ( )FRAC	181
05.05 – قانون الأولوية (التدرج) في تنفيذ العلامات	186
06.05 – ترجمة العبارات الحسابية إلى لغة الـ PASCAL	193
07.05 – عمليات العلاقات و علامات العلاقات	193
07.05 – العمليات المنطقية و العلامات المنطقية	194
تمارين الفصل	195
الفصـــل الســادس: معالجـــة المعطيـــات سلسلـــة أحـــرف	**222**
01.06 – تمهيد	223
02.06 – تغيير معطية من النوع سلسلة أحرف.	223
03.06 – ربط المعطيات من النوع سلسلة أحرف	227
01.03.06 – الربط باستخدام العلامة + ...	229
02.03.06 – الربط باستخدام الدالة ( )Concat	239
04.06 – طول معطية سلسلة أحرف الدالة ( )LENGTH	248
05.06 – استخلاص مجموعة جزئية الدالة ( )COPY	253
06.06 – حذف مجموعة جزئية الدالة ( )DELETE	260
07.06 – الدالة ( )INSERT	262
08.06 – الدالة ( )Ord	264
09.06 – الدالة ( )Chr	270
10.06 – الدالة ( )Succ	273
11.06 – الدالة ( )Pred	275
تمارين الفصل	278
الفصـــل الســابـع: نشـــر (نســـخ) المعطيـــات	**305**
01.07 – تمهيد	306
02.07 – التعليمة Write	306
03.07 – التعليمة Writeln	311

04.07 – أشكال النشر	314
01.04.07 – أشكال نشر المعطيات سلسلة أحرف	314
02.04.07 – أشكال نشر المعطيات (القيم) الصحيحة	317
03.04.07 – أشكال نشر المعطيات (القيم) الحقيقية	321
05.07 – الفرق بين التعليمتين Write و Writeln	324
06.07 – التعليمتان Write و Writeln مع التعليق	326
07.07 – التعليمة Writeln و قفز (تخطي) السطر	331
08.07 – التعليمتان Write و Writeln و نشر نتائج العمليات الحسابية	334
09.07 – التعليمتان Write و Writeln و نشر التعاليق و محتوى المتغيرات و نتائج العمليات الحسابية	337
تمارين الفصل	342
الفصـــل الثامـــن: التعليمـــات الشرطيـــة	**363**
01.08 – تمهيد	364
02.08 – تعليمة القفز اللاشرطي *GOTO*.	364
03.08 – التعليمة الشرطية ....IF....THEN....ELSE	371
04.08 – كتلة تعليمات أو التعليمة المركبة	380
05.08 – التعليمات IF المتداخلة	381
06.08 – التوجيه المتعدد: التعليمة CASE	384
تمارين الفصل	388
الفصـــل التاســـع: المعالجـــة التكراريـــة–الحلقـــات–	**411**
01.09 – تمهيد	412
02.09 – المعالجة التكرارية: الحلقة	413
03.09 – الحلقة بعداد	415
04.09 – الخطوة	422
05.09 – الحلقة *For ... To ... Do*...	424
06.09 – الحلقة *Repeat ... until*	437
07.09 – الحلقة *While ... do*	440

08.09 – الحلقات المتداخلة	443
تمارين الفصل	447
الفصـــل العاشـــر: الجـــداول وحيـــدة البعـــد (الأشعـــة)	**469**
01.10 – تمهيد	470
02.10 – التعريف بالجداول وحيدة البعد (الشعاع)	470
03.10 – بعد الجداول وحيدة البعد (الشعاع)	471
04.10 – التصريح بالجداول وحيدة البعد (الشعاع)	471
05.10 – إدخال المعطيات في الجداول وحيدة البعد (الشعاع)	473
01.05.10 – إدخال المعطيات عن طريق البرنامج	473
02.05.10 – إدخال المعطيات عن طريق لوحة المفاتيح	478
06.10 – العمليات على الجداول وحيدة البعد (الشعاع)	481
01.06.09 – جمع شعاعين	481
02.06.09 – فرق شعاعين	485
03.06.09 – جداء شعاعين	486
07.09 – فرز الجداول وحيدة البعد (الشعاع)	487
01.07.09 – البحث عن أكبر قيمة	487
02.07.09 – البحث عن أقل قيمة	489
تمارين الفصل	491
الفصـــل الحـــادي عشـــر: الجـــداول ثنائيـــة البعـــد (المصفـــوفـــات)	**511**
01.11 – تمهيد	512
02.11 – التعريف بالجداول ثنائية البعد (المصفوفة)	512
03.11 – بعد الجداول ثنائية البعد (المصفوفة)	514
04.11 – التصريح بالجداول ثنائية البعد (المصفوفة)	515
05.11 – إدخال المعطيات في الجداول ثنائية البعد (المصفوفة)	517
01.05.11 – إدخال المعطيات عن طريق البرنامج	517
02.05.11 – إدخال المعطيات عن طريق لوحة المفاتيح	525
06.11 – العمليات على الجداول ثنائية البعد (المصفوفة)	529

01.06.11 – جمع مصفوفتين	529
02.06.11 – فرق مصفوفتين.	533
03.06.11 – جداء مصفوفتين.	534
تمارين الفصل	538
المراجع	545

مقدمة:

المطبوعة التي بين يديك و المعنونة بـ: **" محـاضرات فـي مقيـاس الاعـلام الآلـي2 "** عبارة عن المولود الخامس ضمن سلسلة من الاصدارات التي تم إصدارها من قبل المؤلف في العديد من المقاييس. تهدف هذه المطبوعة إلى تدعيم ما هو متواجد من مراجع في هذا المقياس. تحتوي المطبوعة على **محـاضرات فـي مقيـاس الاعـلام الآلـي2** مدعمة بأمثلة محلولة و معلق عليها موزعة على أحدى عشر فصلا:

الفصـل الأول : مدخل إلى لغة البرمجة PASCAL
الفصـل الثانـي: مفهوم المعطيات: المتغيرات و الثوابت
الفصـل الثالـث: التصريح بالمعطيات
الفصـل الرابـع: إدخال المعطيات
الفصـل الخامس: معالجة المعطيات العددية
الفصـل السادس: معالجة المعطيات سلسلة أحرف
الفصـل السابـع: نشر المعطيات
الفصـل الثامـن: التعليمات الشرطية
الفصـل التاسـع: المعالجة التكرارية – الحلقات -
الفصـل العاشـر: الجداول وحيدة البعد (الأشعة)
الفصـل الحادي عشر: الجداول ثنائية البعد (المصفوفات)

هذه المطبوعة موجهة لطلبة السنة الثانية التخصصات التالية:

- **العلـــوم الاقتصـاديـة**
- **العلـــوم التجـــاريـة**
- **علـــــوم التسييـــــر**
- **علـوم ماليـة و محاسبـة**

هذا العمل المتواضع هو ثمرة تجربة متواضعة من تدريس هذا المقياس لعدة سنوات. في الأخير لا يسعني إلا أن أطلب من أي متصفح لهذه المطبوعة أستاذا أو طالبا أن لا يبخل علينا بأي ملاحظة أو انتقاد حول محتوى هذه المطبوعة.

تيـارت فـي جـانفـي 2019

المؤلـف

الفصـــل 📖 الأول

مـدخل الى لـغة البرمجة PASCAL

Introduction au langage de programmation PASCAL

أهـداف الفصـل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بمـا يلـي:

- بنية و مكونات برنامج بلغة الـ PASCAL .
  1. L'en-tête
  2. جزء التصريح
  3. جزء التعليمات
- Les Identficateurs
- المعرفة مسبقا Les Identificateurs Prédifinies
- الكلمات المحجوزة Les Mots Clés Réservés
- التعاليق Les Commentaires

01.01 – تمهيد:

إضافة إلى لغات البرمجة مثل .... , Basic , Fortran هناك لغة برمجة أخرى تسمى **لغة الباسكال Langage PASCAL** و التي نحاول من خلال هذا الفصل إلى التعريف و إعطاء لمحة عنها على أن يتم الإلمام بها من خلال الفصول اللاحقة.

02.01– أول برنامج بلغة الـ PASCAL: Premier Programme En PASCAL

سوف نتناول من خلال هذه الفقرة عرض أول برنامج بلغة الـ **PASCAL** و الذي سيتم التعليق عليه في الفقرات اللاحقة لهذا الفصل

```
program SOMME_DES_NOMBRES;
const c = 150 ;
var a :integer ;
    b :integer ;
    d,e,s : real ;
    f : char ;
    t : string ;
begin
    writeln ( 'VOILA UN PREMIER PROGRAMME PASCAL ' ) ;
    writeln ( ' JE VOUS CALCULE LA SOMME DE CINQ NOMBRES ');
    {ITRODUCTION DES VALEURS DES VARIABLES a,b,d,e } ;
    b := 5 ;
    e :=10.02 ;
    write ('DONNEZ LA VALEUR DE LA VARIABLE ENTIERE a : ');
    readln (a);
    write ('DONNEZ LA VALEUR DE LA VARIABLE REELLE d : ');
    readln (d);
    write ('DONNEZ LA VALEUR DE LA VARIABLE CARACTERE f : ');
    readln (f);
    write ('DONNEZ LA VALEUR DE LA VARIABLE CHAINE CARACTERES t : ');
    readln (t);
    {CALCUL DE LA SOMME } ;
    s := a +b+c+ d + e ;
    {AFFICHAGE DE LA SOMME } ;
    writeln ('LA SOMME = ',s);
    writeln ('LE CARACTERE INTRODUIT EST : ',f);
    writeln ('LA CHAINE DE CARACTERES INTRODUITE EST : ',t);
end.
```

03.01 – بنية (هيكل) برنامج بلغة الـ PASCAL: Structure d'un Programme

مثله مثل أي برنامج بلغة الـ PASCAL، فإن البرنامج الأول أعلاه يتكون من العناصر أو الأجزاء التالية:

1. L'en-tête
2. جزء التصريح La partie déclaration
3. جزء التعليمات La partie instructions

سوف يتم التطرق بالتفصيل إلى كل جزء على حدى

L'en-tête – 01.03.01

تتمثل L'en-tête للبرنامج الأول أعلاه في التعليمة التالية:

program** SOMME_DES_NOMBRES **;

حيث أن:

- ***program*** : يمثل Un mot clé أو Un mot réservé يسمح بكتابة L'en-tête برنامج
- *SOMME_DES_NOMBRES* : يمثل الاسم المراد إعطائه إلى البرنامج، هذا الاسم ليس له أي معنى بالنسبة للغة البرمجة PASCAL و ليس له أي علاقة بالملف الذي يتم فيه تخزين البرنامج. يفضل أن يكون هذا الاسم له دلالة أو يعبر عن محتوى أو الغاية من البرنامج.
- **;** : تمثل فاصلة منقوطة (point-virgule) و هي إجبارية تقوم بفصل L'en-tête عن باقي تعليمات البرنامج

نشير إلى أن L'en-tête إجبارية في لغة البرمجة PASCAL

02.03.01 – جزء التصريح: La partie déclaration

يمتد جزء التصريح من أول سطر يلي L'en-tête إلى غاية آخر سطر يسبق ***begin*** الأولى و عليه فإن الجزء الخاص بالتصريح للبرنامج الأول أعلاه يتمثل في الستة أسطر (تعليمات) التالية:

```
const c = 150 ;
var a :integer ;
    b :integer ;
    d,e,s : real ;
    f : char ;
    t : string ;
```

جزء التصريح أعلاه يتضمن نوعين من التصريحات هما:

1. التصريح بالثوابت: و الذي يتم بواسطة Le mot clé أو Le mot réservé ***const*** من خلال السطر التالي:

```
const c = 150 ;
```

حيث أن:

- *const* : يمثل Le mot clé أو Le mot réservé الذي يسمح أو يستخدم للتصريح بالثوابت و هو مختصر للكلمة الإنجليزية constant.
- *c* : تمثل اسم الثابتة Identificateur de constante المصرح بها.
- *150* : تمثل قيمة الثابتة التي تحمل الاسم *c*.
- **;** : تمثل فاصلة منقوطة (point-virgule) و هي إجبارية.

فمن خلال السطر أعلاه تم التصريح بالثابتة التي تحمل الإسم *c* و التي تساوي القيمة *150* .

نشير إلى أنه يمكن التصريح بالعديد من الثوابت كما يلي:

```
const c = 150 ;
      g=1234 ;
      r=456 ;
```

2. التصريح بالمتغيرات: و الذي يتم بواسطة Le mot clé أو Le mot réservé ***var*** من خلال الأسطر التالية:

```
var a :integer ;
    b :integer ;
    d,e,s : real ;
    f : char ;
    t : string ;
```

حيث أن:

- *var* : يمثل Le mot clé أو Le mot réservé الذي يسمح أو يستخدم للتصريح بالمتغيرات و هو مختصر للكلمة الإنجليزية varibale.
- *a* : تمثل اسم المتغيرة Identificateur de variable المصرح بها.
- *integer* : يمثل Le mot clé أو Le mot réservé الذي يسمح أو يستخدم للتصريح بالمتغيرات من النوع **الصحيح** Entière.
- *real* : يمثل Le mot clé أو Le mot réservé الذي يسمح أو يستخدم للتصريح بالمتغيرات من النوع **الحقيقي** Réelle.

- *d,e,s* : تمثل أسماء لثلاث متغيرات Trois identificateurs de variables من النوع الحقيقي المصرح بها.
- *char* : يمثل Le mot clé أو Le mot réservé الذي يسمح أو يستخدم للتصريح بالمتغيرات من النوع **الحرف** Caractère.
- *f* : تمثل اسم المتغيرة Identificateur de variable من النوع حرف المصرح بها.
- *string* : يمثل Le mot clé أو Le mot réservé الذي يسمح أو يستخدم للتصريح بالمتغيرات من النوع **سلسلة أحرف** Chaine de caractères.
- *t* : تمثل اسم المتغيرة Identificateur de variable من النوع سلسلة أحرف المصرح بها.

فمن خلال الأسطر التالية:

- *var a :integer;* يتم التصريح بمتغيرة من النوع الصحيح تحمل الاسم *a* .
- *b :integer;* يتم التصريح بمتغيرة من النوع الصحيح تحمل الاسم *b* .
- *d,e,s : real ;* يتم التصريح بثلاث متغيرات من النوع الحقيقي تحمل الأسماء *d,e,s* .
- *f : char ;* يتم التصريح بمتغيرة من النوع حرف تحمل الاسم *f* .
- *t : string ;* يتم التصريح بمتغيرة من النوع سلسلة أحرف تحمل الاسم *t* .

03.03.01 – جزء التعليمات: La partie instructions

يسمى هذا الجزء كذالك بـ La partie exécutable و الذي يمتد من السطر الأول الذي يلي ***begin*** الأولى إلى غاية السطر الأخير الذي يسبق ***end*** الأخيرة و عليه فإن جزء التعليمات للبرنامج الأول أعلاه يتمثل في الأسطر (التعليمات) التالية:

```
writeln ( 'VOILA UN PREMIER PROGRAMME PASCAL ' ) ;
writeln ( ' JE VOUS CALCULE LA SOMME DE CINQ NOMBRES ');
{ITRODUCTION DES VALEURS DES VARIABLES a,b,d,e } ;
b := 5 ;
e :=10.02 ;
write ('DONNEZ LA VALEUR DE LA VARIABLE ENTIERE a : ');
readln (a);
write ('DONNEZ LA VALEUR DE LA VARIABLE REELLE d : ');
readln (d);
write ('DONNEZ LA VALEUR DE LA VARIABLE CARACTERE f : ');
readln (f);
write ('DONNEZ LA VALEUR DE LA VARIABLE CHAINE CARACTERES t : ');
readln (t);
{CALCUL DE LA SOMME } ;
```

```
s := a +b+c+ d + e ;
{AFFICHAGE DE LA SOMME } ;
writeln ('LA SOMME = ',s);
writeln ('LE CARACTERE INTRODUIT EST : ',f);
writeln ('LA CHAINE DE CARACTERES INTRODUITE EST : ',t);
```

نلاحظ أن جزء التعليمات يتضمن العديد من التعليمات هي:

- ***writeln (' JE VOUS CALCULE LA SOMME DE CINQ NOMBRES ')*** :

نعلم أن *write* تعني كتابة (Ecrire) و هي متبوعة بـ *ln* و التي هي مختصر للكلمة الإنجليزية Ligne. ***write*** تطلب من الحاسوب كتابة أو نشر على الشاشة ما هو موجود بين ' ' Entre deux quotes (Apostrophes) أي نشر على الشاشة *JE VOUS CALCULE LA SOMME DE CINQ NOMBRES* . ***ln*** تطلب من الحاسوب بالانتقال إلى السطر الموالي بعد الانتهاء من نشر على ما هو موجود بين ' ' .

- ***writeln (' JE VOUS CALCULE LA SOMME DE CINQ NOMBRES ')*** :

يقوم الحاسوب بنشر على الشاشة *JE VOUS CALCULE LA SOMME DE CINQ NOMBRES* ثم يقوم بعد ذالك بالعودة إلى السطر.

- ***b := 5*** : يقوم الحاسوب بتوجيه و تخزين القيمة الصحيحة 5 إلى المتغيرة من النوع الصحيح *b* التي تم التصريح بها في الجزء الخاص بالتصريح.
- ***e :=10.02*** : يقوم الحاسوب بتوجيه و تخزين القيمة الصحيحة *10.02* إلى المتغيرة الحقيقية *e* التي تم التصريح بها في الجزء الخاص بالتصريح.
- ***write ('DONNEZ LA VALEUR DE LA VARIABLE ENTIERE a : ')*** :

يقوم الحاسوب بنشر على الشاشة

DONNEZ LA VALEUR DE LA VARIABLE ENTIERE a :

دون العودة إلى السطر.

- ***readln (a)*** :

نعلم أن *read* تعني قراءة (lecture) و هي متبوعة بـ *ln* و التي هي مختصر للكلمة الإنجليزية Ligne. ***read*** تطلب من الحاسوب قراءة قيمة المتغيرة الصحيحة ***a*** التي سوف يتم إدخالها عن طريق

لوحة المفاتيح (Le clavier) ليقوم الحاسوب بتخزينها في المتغيرة ***a***. ***ln*** تطلب من الحاسوب بالانتقال إلى السطر الموالي بعد قراءة قيمة المتغيرة الصحيحة ***a***.

- ***write ('DONNEZ LA VALEUR DE LA VARIABLE REELLE d : ')*** :

يقوم الحاسوب بنشر على الشاشة

DONNEZ LA VALEUR DE LA VARIABLE REELLE d :

دون العودة إلى السطر.

- ***readln (d)*** :

read تطلب من الحاسوب قراءة قيمة المتغيرة من النوع الحقيقي ***d*** التي سوف يتم إدخالها عن طريق لوحة المفاتيح (Le clavier) ليقوم الحاسوب بتخزينها في المتغيرة ***d***. ***ln*** تطلب من الحاسوب الانتقال إلى السطر الموالي بعد قراءة قيمة المتغيرة الحقيقية ***d***.

- ***write ('DONNEZ LA VALEUR DE LA VARIABLE CARACTERE f : ')*** :

يقوم الحاسوب بنشر على الشاشة

DONNEZ LA VALEUR DE LA VARIABLE CARACTERE f :

دون العودة إلى السطر.

- ***readln (f)*** :

read تطلب من الحاسوب قراءة قيمة المتغيرة من النوع حرف ***f*** التي سوف يتم إدخالها عن طريق لوحة المفاتيح (Le clavier) ليقوم الحاسوب بتخزينها في المتغيرة ***f***. ***ln*** تطلب من الحاسوب الانتقال إلى السطر الموالي بعد قراءة قيمة المتغيرة ***f***.

- ***write ('DONNEZ LA VALEUR DE LA VARIABLE CHAINE CARACTERES t : ')*** :

يقوم الحاسوب بنشر على الشاشة

DONNEZ LA VALEUR DE LA VARIABLE CHAINE CARACTERES t :

دون العودة إلى السطر.

- ***readln (t)*** :

read تطلب من الحاسوب قراءة قيمة المتغيرة من النوع سلسلة أحرف ***t*** التي سوف يتم إدخالها عن طريق لوحة المفاتيح (Le clavier) ليقوم الحاسوب بتخزينها في المتغيرة ***t***. ***ln*** تطلب من الحاسوب الانتقال إلى السطر الموالي بعد قراءة قيمة المتغيرة سلسلة أحرف ***t***.

- ***s := a+b+c+ d+e*** : تسمى تعليمة التوجيه Instruction d'affectation تقوم هذه التعليمة بإنجاز ما يلي:
  - حساب العملية الحسابية الواقعة على يمين **:=** أي حساب ***a+b+c+ d+e***
  - توجيه (l'affectation) و تخزين نتيجة العملية الحسابية في المتغيرة من النوع الحقيقي *s*
- ***writeln ('LA SOMME = ',s)*** :

بوجود ***write*** يقوم الحاسوب بنشر على الشاشة و على نفس السطر معلومتين هما:

- ما هو موجود بين ' ' و هو **LA SOMME =**
- قيمة أو محتوى المتغيرة من النوع الحقيقي ***s***

ليقوم بعد ذالك بالانتقال إلى السطر الموالي لوجود ***ln***

- ***writeln ('LE CARACTERE INTRODUIT EST : ',f)*** :

بوجود ***write*** يقوم الحاسوب بنشر على الشاشة و على نفس السطر معلومتين هما:

- ما هو موجود بين ' ' و هو ***LE CARACTERE INTRODUIT EST :***
- قيمة أو محتوى المتغيرة من النوع حرف ***f***

ليقوم بعد ذالك بالانتقال إلى السطر الموالي لوجود ***ln***

- ***writeln ('LA CHAINE DE CARACTERES INTRODUITE EST : ',t)*** :

بوجود ***write*** يقوم الحاسوب بنشر على الشاشة و على نفس السطر معلومتين هما:

- ما هو موجود بين ' ' و هو ***LA CHAINE DE CARACTERES INTRODUITE EST :***
- قيمة أو محتوى المتغيرة من سلسلة أحرف ***t***

ليقوم بعد ذالك بالانتقال إلى السطر الموالي لوجود ***ln***

:Les Identificateurs – 04.01

بتفحص البرنامج الأول بلغة الـ PASCAL نلاحظ العديد من الكلمات (Les mots) المستخدمة من قبل مصمم أو معد البرنامج مثل *a,b,c,d,t,f* . هذه الكلمات المستخدمة و التي تمثل أسماء ثوابت أو متغيرات بأنواعها تسمى في لغة البرمجة PASCAL بـ: Les Identificateurs و

التي تم استخدامها لتعيين الثوابت و المتغيرات. مصمم البرنامج حر في اختيار Les Identificateurs شريطة احترام القواعد التالية:

1. الحروف المسموح بها لتشكيل و تكوين Les Identificateurs هي:
   - الحروف اللاتينية (A,B,C,…..,Z)
   - الأعداد (0,1,2,3,….,9)
   - الحرف _ Le Caractère Souligné .
2. الحرف الأول يجب أن يكون حرف من الحروف (A,B,C,…..,Z)
3. يمكن لـ: _ Le Caractère Souligné أن يكون في أول Les Identificateurs و لكن لا ينصح بذالك.
4. الحاسوب لا يفرق بين الأحرف الكبيرة Les Lettres Majuscules و الأحرف الصغيرة Les Lettres Miniscules .
5. أن لا تحتوي على أي حرف خاص من الأحرف الخاصة Caractères Spéciaux
6. أن لا تحتوي على أي فراغ Espace .
7. يمكن أن تتضمن العديد من الأحرف أي ذات طول مهما كان، غير أن لغة البرمجة PASCAL لا تأخذ بعين الاعتبار إلا الثمانية أحرف الأولى.

بغرض توضيح أكثر لـ Les Identificateurs و قواعد اختيارها نأخذ المثال التالي:

مثال 01.01:

- *5p* : خاطئ لأنه يبدأ بـ عدد
- *so m* : خاطئ لأنه يتضمن أو يحتوي على فراغ Espace
- cté : لأنه يحتوي على Un Caractère Accenté
- x : صحيح
- x2 : صحيح
- a_1 : صحيح
- rt0_56 : صحيح
- _zE6 : صحيح
- _5 : صحيح
- x هو X نفسه لأن الحاسوب لا يفرق بين Majuscules و Miniscules .
- valeur هو VAleuR نفسه لأن الحاسوب لا يفرق بين Majuscules و Miniscules .

ملاحظة 01.01 :

لا يمكن استخدام Un Identificateur ما لم يتم التصريح به

05.01 – المعرفة مسبقا: Les Identificateurs Prédéfinies

إضافة إلى Les Identificateurs المستخدمة أو المستعملة من قبل مصمم البرنامج و التي تمثل أسماء المتغيرات و الثوابت بجميع أنواعها، هناك نوع آخر من Les Identificateurs و التي تسمى معرفة مسبقا Les Identificateurs Prédéfinies و تسمى كذالك Les Mot Clés Standards ، هذه الأخيرة لها هدف محدد و مسبق في لغة البرمجة PASCAL ، و لكن ليست محجوزة مثل Les Mot Clés Réservés أي أنه يمكن لمصمم البرنامج استخدامها كأسماء ثوابت أو متغيرات. من بين Les Identificateurs Prédéfinies نذكر ما يلي:

- FALSE , TRUE , MAXINT
- INTEGER, BOOLEAN, REAL, CHAR, TEXT
- INPUT , OUTPUT
- ABS,ARCTAN,CHR,COS,EOF,EOLN,EXP,LN,ODD
- ORD,PRED,ROUND,SIN,SQR,SQRT,SUCC,TRUNC
- GET,NEW,PACK,PUT,READ,READLN,RESET
- REWRITE,UNPACK,WRITE,WRITELN

ملاحظة 02.01 :

إذا تم استخدام معرفة مسبقا Un Identificateurs Prédéfinies كا Identificateur ، فإنه لا يمكن إعادة استخدامه من أجل غرضه الأصلي. فمثلا إذا تم استخدام الدالة ***sqr*** كاسم لمتغيرة أو ثابتة فإنه لا يمكن إعادة استخدامها من أجل غرضها الأصلي الذي يتمثل في إيجاد أو حساب المربع. لذلك ينصح أن لا يتم استخدام المعرفة مسبقا Un Identificateurs Prédéfinies كا Identificateur أي كأسماء لثوابت أو متغيرات

06.01 – الكلمات المحجوزة: Les Mots Clés Réservés

إضافة إلى كل من Les Identificateurs و Les Identificateurs Prédéfinies ، هناك نوع آخر يسمى الكلمات المحجوزة Les Mot Clés Réservés ، هذه الأخيرة محجوزة من قبل لغة البرمجة PASCAL لها هدف محدد أي لها معنى خاص في لغة البرمجة PASCAL ، حيث لا يمكن أبد لمصمم البرنامج استخدامها كأسماء ثوابت أو متغيرات. الكلمات المحجوزة هي:

- AND,ARRAY,BEGIN,CASE,CONST,DIV,DO,DOWNT

- ELSE,END,FILE,FOR,FUNCTION,GOTO,IF,IN,LABEL
- MOD,NIL,NOT,OF,OR,PACKED,PROCEDURE,PROGRAM
- RECORD,REPEAT,SET,THEN,TO,TYPE,UNTIL,VAR
- WHILE,WITH

07.01 – الشكل الحر Le Format Libre:

بتفحص البرنامج الأول نلاحظ أن كل سطر يحتوي على تعليمة واحدة فقط أي كل تعليمة تم كتابتها في سطر واحد. على عكس بعض لغات البرمجة الأخرى، فإن لغة البرمجة PASCAL تسمح بكتابة العديد من التعليمات في نفس السطر هذه الطريقة في كتابة البرنامج تسمى بالشكل الحر Format Libre و عليه فإن البرنامج الأول يمكن كتابته على الشكل التالي:

```
program SOMME_DES_NOMBRES;const c = 150 ;var a :integer ;b :integer ;
    d,e,s : real ;f : char ;t : string ;begin
    writeln ( 'VOILA UN PREMIER PROGRAMME PASCAL ' ) ;
    writeln ( ' JE VOUS CALCULE LA SOMME DE CINQ NOMBRES ');
    {ITRODUCTION DES VALEURS DES VARIABLES a,b,d,e } ;  b := 5 ;
    e :=10.02 ;  write ('DONNEZ LA VALEUR DE LA VARIABLE ENTIERE a : ');
    readln (a);  write ('DONNEZ LA VALEUR DE LA VARIABLE REELLE d : ');
    readln (d);  write ('DONNEZ LA VALEUR DE LA VARIABLE CARACTERE f : ');
    readln (f);
    write ('DONNEZ LA VALEUR DE LA VARIABLE CHAINE CARACTERES t : ');
    readln (t);  {CALCUL DE LA SOMME } ;  s := a +b+c+ d + e ;
    {AFFICHAGE DE LA SOMME } ;  writeln ('LA SOMME = ',s);
    writeln ('LE CARACTERE INTRODUIT EST : ',f);
    writeln ('LA CHAINE DE CARACTERES INTRODUITE EST : ',t); end.
```

ملاحظـة 03.01 :

عندما يتم كتابة العديد من التعليمات في نفس السطر، فإن البرنامج يبدو غير واضح. لذلك ينصح بكتابة البرنامج بهذا الشكل الحر

08.01 – التعاليق Les Commentaires:

مثل غيرها من لغات البرمجة، فإن لغة البرمجة PASCAL تسمح بوجود أو إدراج تعاليق ضمن البرنامج. هذه التعاليق عبارة عن نص تفسيري أو توضيحي موجه لقارئ البرنامج. يتم كتابة هذه التعاليق في البرنامج عن طريق وضعها بين حاضنتين **{ }** أو الرمزين **(* *)** كما يلي:

{ COMMENTAITE **} ;**

(* COMMENTAITE *) ;

البرنامج الأول يتضمن العديد من التعاليق هي:

- { ITRODUCTION DES VALEURS DES VARIABLES a,b,d,e}

و الذي يوضح لقارئ البرنامج بأن التعليمات التي تلي هذا التعليق خاصة بإدخال قيم المتغيرات a,b,d,e

- {CALCUL DE LA SOMME }

و الذي يوضح بأن التعليمات التي تلي هذا التعليق خاصة بحساب المجموع

- {AFFICHAGE DE LA SOMME }

و الذي يوضح بأن التعليمات التي تلي هذا التعليق خاصة بنشر المجموع

تمارين الفصل

✍ التمرين 01.01:

1. قدم مختلف الأجزاء و المكونات التي يتكون منها برنامج بلغة الـ PASCAL

✍ التمرين 02.01:

1. ماذا نعني بـ Les Identificateurs
2. ما هي القواعد الواجب إتباعها في كتابة Les Identificateurs

✍ التمرين 03.01:

لتكن Les Identificateurs التالية:

- *5p*
- *so m*
- cté
- x
- x2
- a_1
- rt0_56
- _zE6
- _5

- **المطلـوب:**

1. بين من بين Les Identificateurs أعلاه الصحيحة و الخاطئة.

✍ التمرين 04.01:

1. ماذا نعني بـ Les Identificateurs Prédéfinies.
2. أذكرها.
3. هل يمكن استخدامها كا Des Identificateurs.

✍ التمرين 05.01:

1. ماذا نعني بـ Les Mots Clés Réservés.
2. أذكرها.

3. هل يمكن استخدامها كا Des Identificateurs.

✍ التمرين 06.01:

1. ماذا نعني بالتعاليق Les Commentaires.
2. قدم طريقة كتابتها.

✍ التمرين 07.01:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program SOMME_DES_NOMBRES;
const c = 150 ;
var a :integer ;
    b :integer ;
    d,e,s : real ;
    f : char ;
    t : string ;
begin
    writeln ( 'VOILA UN PREMIER PROGRAMME PASCAL ' ) ;
    writeln ( ' JE VOUS CALCULE LA SOMME DE CINQ NOMBRES ');
    b := 5 ;
    e :=10.02 ;
    write ('DONNEZ LA VALEUR DE LA VARIABLE ENTIERE a : ');
    readln (a);
    write ('DONNEZ LA VALEUR DE LA VARIABLE REELLE d : ');
    readln (d);
    write ('DONNEZ LA VALEUR DE LA VARIABLE CARACTERE f : ');
    readln (f);
    write ('DONNEZ LA VALEUR DE LA VARIABLE CHAINE CARACTERES t : ');
    readln (t);
    s := a +b+c+ d + e ;
    writeln ('LA SOMME = ',s);
    writeln ('LE CARACTERE INTRODUIT EST : ',f);
    writeln ('LA CHAINE DE CARACTERES INTRODUITE EST : ',t);
end.
```

- **المطلوب:**

1. إدراج تعاليق ضمن البرنامج أعلاه حتى يتسنى لنا معرفة التعليمات الخاصة بكل من التصريح و إدخال المعطيات و العمليات الحسابية و نشر المعطيات.

✍ التمرين 08.01:

1. هل يشترط في لغة البرمجة PASCAL أن يتم كتابة كل تعليمة في سطر واحد
2. هل يمكن كتابة العديد من التعليمات في سطر واحد

✍ التمرين 09.01:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program SOMME_DES_NOMBRES;const c = 150 ;var a :integer ;b :integer ;
    d,e,s : real ;f : char ;t : string ;begin
    writeln ( 'VOILA UN PREMIER PROGRAMME PASCAL ' ) ;
    writeln ( ' JE VOUS CALCULE LA SOMME DE CINQ NOMBRES ');
    {ITRODUCTION DES VALEURS DES VARIABLES a,b,d,e } ;  b := 5 ;
    e :=10.02 ;  write ('DONNEZ LA VALEUR DE LA VARIABLE ENTIERE a : ');
    readln (a);  write ('DONNEZ LA VALEUR DE LA VARIABLE REELLE d : ');
    readln (d);  write ('DONNEZ LA VALEUR DE LA VARIABLE CARACTERE f : ');
    readln (f);
    write ('DONNEZ LA VALEUR DE LA VARIABLE CHAINE CARACTERES t : ');
    readln (t);  {CALCUL DE LA SOMME } ;  s := a +b+c+ d + e ;
    {AFFICHAGE DE LA SOMME } ;  writeln ('LA SOMME = ',s);
    writeln ('LE CARACTERE INTRODUIT EST : ',f);
    writeln ('LA CHAINE DE CARACTERES INTRODUITE EST : ',t); end.
```

- **المطلوب:**

1. إعادة كتابة البرنامج أعلاه حتى يصبح أكثر وضوحا.

✍ التمرين 10.01:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program XXXX ;
const
a=
12;
var  e
:real
; begin write ('donner la valeur de e:');
readln
(e); write ( 'la valeur introduite est :',e); end.
```

- **المطلـوب:**

1. إعادة كتابة البرنامج أعلاه حتى يصبح أكثر وضوحا.

الفصل 📖 الثاني

مفهوم المعطيات
المتغيرات و الثوابت

Notion De Données

Les Variables et Les Constantes

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- المعطيات
- أنواع المعطيات.
  1. المعطيات الصحيحة.
  2. المعطيات الحقيقية.
  3. المعطيات سلسلة أحرف.
  4. المعطيات من النوع حرف.
- أنواع المتغيرات.
- المتغيرات.
  5. المتغيرات الصحيحة.
  6. المتغيرات الحقيقية.
  7. المتغيرات سلسلة أحرف.
  8. المتغيرات من النوع حرف.
- الثوابت

01.02 – تمهيد:

الحاسوب أو الكمبيوتر عبارة عن حاسب عددي يقوم أولا بتخزين المعلومات التي تقدم له من خلال البرنامج ليقوم بعد ذالك بمعالجتها ([1]). هذه المعلومات تسمى معطيات Des Données. فمثلا إذا أردنا إجراء عملية حسابية و لتكن على سبيل المثال جمع الأعداد 150 و 12 و 35.86 فإن المعلومات في هذا المثال تتمثل في العددين الصحيحين 150 و 12 العدد الحقيقي 35.86، هذه الأعداد تسمى معطيات. سوف يتم التطرق من خلال هذا الفصل إلى التعريف بالمعطيات و بأنواعها.

02.02 – المعطيات: Les Données

تعرف المعطيات على أنها تلك المعلومات التي يتم إدخالها إلى الحاسوب عن طريق البرنامج حيث يقوم بتخزينها على مستوى الذاكرة على أن يقوم بمعالجتها بعد ذالك. هذه المعطيات تتكون من إحدى العناصر أدناه أو تركيبة Combinaison منها

1. الأحرف العددية Les Caractères Numérique و التي تتمثل في الأعداد 0,1,2,....,9 .
2. الأحرف اللاتينية (A,B,C,….,Z) و (a,b,c,d,….,z) .
3. الأحرف الخاصة Les Caractères Spéciaux: تتمثل في ما يلي:
   - الفراغ L'espace.
   - + , - , *, / , = , < , > , (,) , [,] , ' , ' , . , : , , , ; , { , }
4. الرموز الخاصة Symboles Spéciaux : تتمثل في ما يلي:

+,-,*,/,=,:=,<=,>=,<>,<,>,(,) , [,] , ' , ' , . , : , , , ; , { , } , , .. ,

مثال 01.02:

من بين المعطيات التي تتعلق بموظف ما هي التالية:

Numéro :	78	الرقم:
Matricule	345/1990	رقم التسجيل
Nom:	Halfay	اللقب:
Prénom:	Sid Ali	الاسم:
Date De Naissance:	17 / 03 / 1971	تاريخ الازدياد:
Sexe	M	الجنس:
Adresse:	Sougueur Tiaret	العنوان:

[1] سوف يتم التعرف في الفصلين الخامس و السادس على ماذا نعني بمعالجة المعطيات

	14101	
Grade:	O.P.2	الرتبة:
Taille:	1.75	الطول:

03.02 – أنواع المعطيات: Type Des Données

انطلاقا من المثال 01.02 أعلاه يمكن أن نميز العديد من المعطيات هي:

- المعطية المتعلقة بالرقم و الممثلة في 78 تتكون فقط من الأحرف العددية أي فقط من الأعداد 7 و 8 و بالتالي هي عبارة عن معطية عددية و بما أن المعطية 78 عبارة عن عدد صحيح فإنها تسمى **معطية صحيحة Donnée Entière** .
- المعطية المتعلقة بالطول و الممثلة في 1.75 عبارة عن عدد حقيقي فإنها تسمى **معطية حقيقية Donnée Réelle** .
- المعطية المتعلقة باللقب و الممثلة في Halfay تتكون فقط من الأحرف اللاتينية أي من الأحرف H ، a ، l ، f ، a و y أي أنها عبارة عن توليفة من الحروف اللاتينية في هذه الحالة هذه المعطية تسمى **معطية سلسلة أحرف (حروف) Donnée Chaine De Caractères** .
- المعطية المتعلقة بالاسم و الممثلة في Sid Ali تتكون من:
  1. الأحرف اللاتينية S ، i ، d ، A ، l و i
  2. الأحرف الخاصة الممثلة في الفراغ espace الذي يفصل بين Sid و Ali

في هذه الحالة المعطية عبارة عن توليفة من الأحرف اللاتينية و الأحرف الخاصة فهي كذالك تسمى **معطية سلسلة أحرف (حروف) Donnée Chaine De Caractères** .

- المعطية المتعلقة بالعنوان و الممثلة في Sougueur Tiaret 14101 تتكون من:
  1. الأحرف اللاتينية: S,o,u,g,u,e,u,r,T,I,a,r,e,t
  2. الأعداد: 1 , 4 , 1 , 0 , 1
  3. الأحرف الخاصة الممثلة في الفراغ espace الذي يفصل بين Sougueur و Tiaret
  4. الأحرف الخاصة الممثلة في الفراغ espace الذي يفصل بين 14101 و Tiaret

في هذه الحالة المعطية عبارة عن توليفة من **الأحرف اللاتينية** و **الأعداد** و **الأحرف الخاصة** فهي كذالك تسمى **معطية من النوع سلسلة أحرف (حروف) Donnée Chaine De Caractères** .

- المعطية المتعلقة بالرتبة و الممثلة في O.P.2 تتكون من:
  1. الأحرف اللاتينية: P, O.
  2. الأحرف الخاصة الممثلة في النقطتين.

3. الأعداد: 2

في هذه الحالة المعطية عبارة عن توليفة من **الأحرف اللاتينية** و **الأعداد** و **الأحرف الخاصة** فهي كذالك تسمى **معطية سلسلة أحرف.**

- المعطية المتعلقة برقم التسجيل و الممثلة في 345/1990 تتكون من:
  1. الأعداد: 3451990
  2. الأحرف الخاصة : /

في هذه الحالة المعطية عبارة عن توليفة من **الأعداد** و **الأحرف الخاصة** فهي كذالك تسمى **معطية سلسلة أحرف.**

- المعطية المتعلقة بتاريخ الازدياد و الممثلة في 1971 / 03 / 17 تتكون من:
  1. الأعداد: 1971 03 17
  2. الأحرف الخاصة : /

في هذه الحالة المعطية عبارة عن توليفة من **الأعداد** و **الأحرف الخاصة** فهي كذالك تسمى **معطية سلسلة أحرف.**

- المعطية المتعلقة بالجنس و الممثلة في M تتكون من حرف لاتيني واحد فقط M. في هذه الحالة المعطية تسمى **معطية من النوع حرف Donnée De Type Caractère** .

من خلال ما تم التطرق إليه أعلاه نميز الأنواع التالية:

1. المعطيات الصحيحة.
2. المعطيات الحقيقية.
3. المعطيات سلسلة أحرف.
4. المعطيات من النوع حرف.

سوف يتم التطرق بالتفصيل إلى كل نوع من خلال الفقرات أدناه:

01.03.02 – المعطيات الصحيحة: Les Données Entières

تتمثل المعطيات الصحيحة في الأعداد الصحيحة السالبة و الموجبة. المثال أدناه يقدم بع ض المعطيات من النوع الصحيح.

مثال 02.02:

المعطيات أدناه تمثل معطيات صحيحة

- 675 – • 15000 • 123 • 2456

- $+768$
- -15
- -3456
- 9087

02.03.02 – المعطيات الحقيقية: Les Données Réelles

تتمثل المعطيات الحقيقية في الأعداد الحقيقية السالبة أو الموجبة المكتوبة بفاصلة عشرية مثل $-123,45$ و $0,0056$ غير أن في لغة البرمجة PASCAL الفاصلة تمثل بنقطة حيث أن:

- $-123,45$ تكتب -123.45
- $0,0056$ تكتب 0.0056

المثال أدناه يقدم بعض المعطيات من النوع الحقيقي المكتوبة بفاصلة عشرية.

مثال 03.02:

المعطيات أدناه تمثل معطيات حقيقية

- -675.009
- 1.57
- 1.23
- 9.01
- $+0.068$
- -34.15
- -34.56
- 0.67

نشير إلى أنه يمكن كتابة الأعداد الحقيقية على الشكل الآسي أي مكون من عدد صحيح أو حقيقي يسمى Mantisse متبوع بالحرف اللاتيني E و بعدد صحيح يسمى Exposant

- $-123,45\times10^{5}$ تكتب $-123,45E5$ أو $-123,45E+5$
- $10,7\times10^{-7}$ تكتب $10,7E-7$

03.03.02 – المعطيات سلسلة أحرف: Les Données Chaine De Caractères

المعطيات سلسلة أحرف عبارة عن توليفة أو تركيبة Combinaison من الأحرف اللاتينية، الأعداد و الأحرف الخاصة طولها لا يتعدى 255 حرف. المثال أدناه يقدم بعض المعطيات من النوع سلسلة أحرف.

مثال 04.02:

المعطيات أدناه تمثل معطيات سلسلة أحرف

- ZAHRA : تركيبة من الحروف اللاتينية.
- TIaret : تركيبة من الحروف.
- RtyAE1235 : تركيبة من الحروف اللاتينية و الأعداد.
- FGop:*/+ : تركيبة من الحروف اللاتينية و الحروف الخاصة.
- 456YUJjk ,:*6- : تركيبة من الحروف اللاتينية و الحروف الخاصة و الأعداد.
- : , / {] ‘ : تركيبة من الحروف الخاصة.
- 345780 : تركيبة من الأعداد.

- 87{+*/56 : تركيبة من الحروف الخاصة و الأعداد.

04.03.02 – المعطيات من النوع حرف: Les Données De Type Caractère

إذا كانت المعطيات من النوع سلسلة حروف عبارة عن مجموعة من الحروف، فإن المعطيات من النوع حرف عبارة حرف واحد فقط و الذي قد يكون حرف لاتيني أو حرف خاص أو عدد. المثال أدناه يقدم بعض المعطيات من النوع حرف.

مثال 05.02:

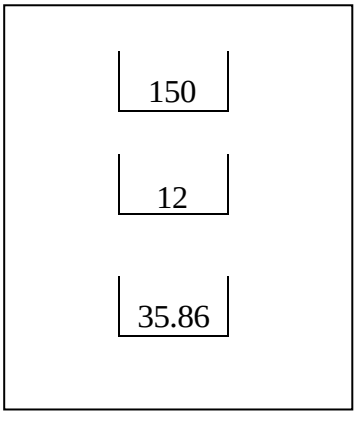
المعطيات أدناه تمثل معطيات من النوع حرف

- r : معطية من النوع حرف تتكون من حرف لاتيني.
- Z : معطية من النوع حرف تتكون من حرف لاتيني.
- / : معطية من النوع حرف تتكون من حرف خاص.
- * : معطية من النوع حرف تتكون من حرف خاص.
- 5 : معطية من النوع حرف تتكون من عدد.
- 6 : معطية من النوع حرف تتكون من عدد.
- : معطية من النوع حرف تتكون من حرف خاص يتمثل في فراغ.

04.02 – المتغيـرات: Les Variables

إذا أردنا جمع المعطيات الممثلة في الأعداد 150 و 12 و 35.86 فإننا نقوم بإدخالها عن طريق البرنامج حيث يقوم الحاسب بتخزينها في الذاكرة ليقوم بعد ذالك بجمعها أي بمعالجتها. عندما يتم تخزين المعطيات في الذاكرة، فإن كل معطية تشغل مكانا أو حيزا على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحية RAM



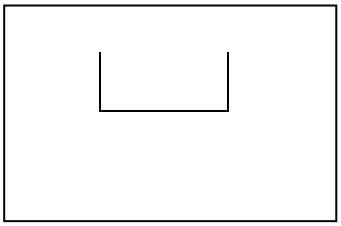
هذا الحيز من الذاكرة الذي يتم فيه تخزين معطية ما يسمى **متغيرة Variable** و بالتالي يمكن استنتاج التعريف التالي:

التعريف 04.02: المتعلق بالمتغيرة

المتغيرة في الإعلام الآلي عبارة حيز من الذاكرة Espace Mémoire أو خانة ذاكرة Case Mémoire يتم فيها تخزين معطية من أنواع المعطيات أعلاه.

يتم تمثيلها على مستوى الذاكرة كما يلي:

الذاكرة الحية RAM

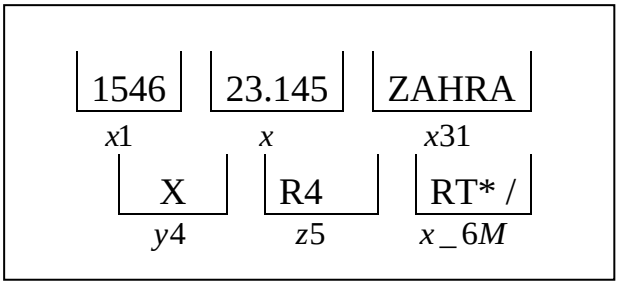


ملاحظـة 01.02 :

كل متغيرة لها اسم هذا الاسم يسمى Identificateur [1]

مثـال 06.02:

الذاكرة الحيـة



- خانة ذاكرة أو المتغيرة التي تحمل الاسم $x1$ تحتوي على المعطية الصحيحة 1546
- خانة الذاكرة أو المتغيرة التي تحمل الاسم x تحتوي على المعطية الحقيقية 23.145
- خانة الذاكرة أو المتغيرة التي تحمل الاسم $x31$ تحتوي على المعطية سلسلة أحرف ZAHRA
- خانة الذاكرة أو المتغيرة التي تحمل الاسم $z5$ تحتوي على المعطية سلسلة أحرف R4
- خانة الذاكرة أو المتغيرة التي تحمل الاسم x_6M تحتوي على المعطية سلسلة أحرف / *RT
- خانة الذاكرة أو المتغيرة التي تحمل الاسم $y4$ تحتوي على المعطية من النوع حرف X

[1] لمزيد من المعلومات حول Les Identificateurs أنظر الفقرة 04.01

ملاحظـة 02.02 :

محتوى خانة ذاكرة (المتغيرة) يمكن أن يتغير أثناء أو خلال تنفيذ البرنامج حيث أن القيمة الجديدة تأخذ مكان القيمة القديمة.

05.02 – أنـواع المتغيـرات: Type Des Variables

يتحدد نوع المتغيرة بنوع المعطية التي تتضمنها هذه المتغيرة، و بما أنه لدينا أربعة أنواع من المعطيات، فإنه لدينا أربعة أنواه من المتغيرات هي:

1. المتغيرات الصحيحة.
2. المتغيرات الحقيقية.
3. المتغيرات سلسلة أحرف.
4. المتغيرات من النوع حرف.

سوف يتم التطرق بالتفصيل إلى كل نوع من خلال الفقرات أدناه:

01.05.02 – المتغيرات الصحيحة: Les Variables Entières

المتغيرة الصحيحة عبارة عن خانة ذاكرة Case Mémoire لا تحتوي إلا على المعطيات الصحيحة، أي عبارة عن حيز من الذاكرة مخصص لتخزين المعطيات من النوع الصحيح فقط.

مثـال 07.02:

المتغيرات أدناه عبارة متغيرات صحيحة

متغيرة صحيحة تحمل الاسم ab_1 | -12356 | ab_1

متغيرة صحيحة تحمل الاسم t | 108754 | t

متغيرة صحيحة تحمل الاسم $x756$ | 78904 | $x756$

02.05.02 – المتغيرات الحقيقية: Les Variables Réelles

المتغيرة الحقيقية عبارة عن خانة ذاكرة Case Mémoire لا تحتوي إلا على المعطيات الحقيقية ، أي عبارة عن حيز من الذاكرة مخصص لتخزين المعطيات من النوع الحقيقي فقط.

مثال 08.02:

المتغيرات أدناه عبارة متغيرات حقيقية

متغيرة حقيقية تحمل الاسم a | -17.567

a

متغيرة حقيقية تحمل الاسم A_1 | 10.098

A_1

متغيرة حقيقية تحمل الاسم YR | 5.768

YR

03.05.02 – المتغيرات سلسلة أحرف: Les Variables Chaine De Caractères

المتغيرة سلسلة أحرف عبارة عن خانة ذاكرة Case Mémoire لا تحتوي إلا على المعطيات سلسلة أحرف، أي عبارة عن حيز من الذاكرة مخصص لتخزين المعطيات من النوع سلسلة أحرف فقط.

مثال 09.02:

المتغيرات أدناه عبارة متغيرات سلسلة أحرف

متغيرة سلسلة أحرف تحمل الاسم CH | $SOUGUEUR$

CH

متغيرة سلسلة أحرف تحمل الاسم $chaine$ | $tiaret14$

$chaine$

متغيرة سلسلة أحرف تحمل الاسم C | $Z.*/:768$

C

ملاحظة 03.02 :

- **يمكن استخدام المتغيرات من النوع سلسلة أحرف لتخزين المعطيات من النوع حرف.**
- **يمكن استخدام المتغيرات من النوع سلسلة أحرف لتخزين المعطيات العددية (الصحيحة و الحقيقية) غير أنه لا يمكن إجراء العمليات الحسابية عليها.**

04.05.02 – المتغيرات من النوع حرف: Les Variables De Type Caractère

المتغيرة من النوع حرف عبارة عن خانة ذاكرة Case Mémoire لا تحتوي إلا على المعطيات من النوع حرف، أي عبارة عن حيز من الذاكرة مخصص لتخزين المعطيات من النوع حرف فقط.

مثال 10.02:

المتغيرات أدناه عبارة متغيرات من النوع حرف

متغيرة سلسلة أحرف تحمل الاسم *H* | *Z* |

متغيرة سلسلة أحرف تحمل الاسم *chaine* | *CH* / *tiaret*14 |

متغيرة سلسلة أحرف تحمل الاسم *C* | *chaine* / *Z*.*/:768 / *C* |

05.05.02 – المتغيرات البولية : Les Variables Booléennes

إضافة إلى الأنواع الأربعة للمتغيرات المذكورة أعلاه، هناك نوع خامس من المتغيرات يسمى **المتغيرات البولية** أو **المتغيرات المنطقية** Les Variables Logiques و التي تأخذ معطيتين أو قيمتين فقط تنفي أو تستبعد إحداهما الأخرى، هذين القيمتين هما:

- true و التي تعني صحيح.
- false و التي تعني خطأ.

06.02 – العمليات على المتغيرات (المعطيات):

إن الغرض الأساسي من إدخال المعطيات بأنواعها هو المعالجة أي إجراء العمليات الحسابية أو المنطقية، حيث أن كل نوع له العمليات الخاصة به. سوف يتم التطرق من خلال هذه الفقرة إلى مختلف العمليات الممكن إجرائها على كل نوع.

01.06.02 – العمليات على المتغيرات (المعطيات) الصحيحة :

تتمثل العمليات الممكن إجراءها على المعطيات من النوع الصحيح في ما يلي:

1. العمليات الحسابية.
2. العمليات العلاقاتية

3. الدوال المعرفة مسبقا.

- **العمليات الحسابية Les Opérations Arithmétiques** [1]

الجدول أدناه يقدم مختلف العمليات الحسابية الممكن إجرائها على المعطيات الصحيحة

العملية الحسابية	الرمز أو المعامل	مثال	نوع النتيجة
الجمع	+	$5+12=17$	معطية صحيحة
الطرح	–	$10-15=-5$	معطية صحيحة
الجداء (الضرب)	*	$25*10=250$	معطية صحيحة
القسمة الحقيقية	/	$1/2=0.5$	معطية حقيقية
الأس	**	$3**4=81$	معطية صحيحة
القسمة الصحيحة	Div	$13 div 5=2$	معطية صحيحة
باقي القسمة الصحيحة	mod	$13 \bmod 5=3$	معطية صحيحة

- **العمليات العلاقاتية Les Opérations De Relation** [2]

الجدول أدناه يقدم مختلف العمليات العلاقاتية الممكن إجرائها على المعطيات الصحيحة

العملية العلاقاتية	الرمز أو المعامل	نوع النتيجة
المساواة	=	معطية Boolienne
اللامساواة	<>	معطية Boolienne
الأكبر	>	معطية Boolienne
الأصغر	<	معطية Boolienne
الأكبر أو يساوي	>=	معطية Boolienne
الأصغر أو يساوي	<=	معطية Boolienne

- **الدوال المعرفة مسبقا Les Fonctions Prédéfinies**

الجدول أدناه يقدم بعض الدوال الممكن إجرائها على المعطيات الصحيحة

[1] لمزيد من التفصيل حول العمليات الحسابية أنظر الفقرة 03.05

[2] لمزيد من التفصيل حول العمليات العلاقاتية أنظر الفقرة 07.05

الدالة	الرمز	مثال	نوع النتيجة
القيمة المطلقة	abs	$abs(-123)=123$	معطية صحيحة
المربع	sqr	$sqr(13)=169$	معطية صحيحة
الجذر التربيعي	sqrt	$sqrt(16)=4$	معطية حقيقية

02.06.02 – العمليات على المتغيرات (المعطيات) الحقيقية :

تتمثل العمليات الممكن إجراءها على المعطيات من النوع الحقيقي في ما يلي:

1. العمليات الحسابية.
2. العمليات العلاقاتية
3. الدوال المعرفة مسبقا.

- **العمليات الحسابية Les Opérations Arithmétiques**

الجدول أدناه يقدم مختلف العمليات الحسابية الممكن إجرائها على المعطيات الحقيقية

العملية الحسابية	الرمز أو المعامل	مثال	نوع النتيجة
الجمع	+	$0.5+1.2=1.7$	معطية حقيقية
الطرح	–	$4.6-2.5=2.1$	معطية حقيقية
الجداء (الضرب)	*	$2.5*1.4=3.5$	معطية حقيقية
القسمة الحقيقية	/	$8.5/2.4=3.541$	معطية حقيقية
الأس	**	$2.5**4.3=6.830$	معطية حقيقية

- **العمليات العلاقاتية Les Opérations De Relation**

الجدول أدناه يقدم مختلف العمليات العلاقاتية الممكن إجرائها على المعطيات الحقيقية

العملية العلاقاتية	الرمز أو المعامل	نوع النتيجة
المساواة	=	معطية Boolienne
اللامساواة	<>	معطية Boolienne
الأكبر	>	معطية Boolienne
الأصغر	<	معطية Boolienne
الأكبر أو يساوي	>=	معطية Boolienne

الأصغر أو يساوي	<=	معطية Boolienne

- **الدوال المعرفة مسبقا Les Fonctions Prédéfinies**

الجدول أدناه يقدم بعض الدوال الممكن إجرائها على المعطيات الصحيحة

الدالة	الرمز	مثال	نوع النتيجة
القيمة المطلقة	abs	$abs(-12.3) = 12.3$	معطية حقيقية
المربع	sqr	$sqr(1.3) = 1.69$	معطية حقيقية
الجذر التربيعي	sqrt	$sqrt(1.6) = 1.26$	معطية حقيقية
الجب	Sin		معطية حقيقية
التجب	Cos		معطية حقيقية
التضل	Actg		معطية حقيقية
اللوغريتم النبيري	Ln		معطية حقيقية
الجزء العشري	frac	$frac(56.67) = 0.67$	معطية حقيقية
الجزء الصحيح	Int	$\text{int}(15.60) = 15$	معطية حقيقية
التقريب أقرب عدد صحيح	Round	$round(2.85) = 3$	معطية صحيحة
الجزء الصحيح	trunc	$trunc(76.94) = 76$	معطية صحيحة

03.06.02 – العمليات على المتغيرات (المعطيات) سلسلة أحرف: [1]

تتمثل العمليات الممكن إجراءها على المعطيات من النوع سلسلة أحرف في ما يلي:

1. تغيير معطية من النوع سلسلة أحرف.
2. ربط المعطيات من النوع سلسلة أحرف.
3. حذف مجموعة جزئية.
4. دمج معطيتين من النوع سلسلة أحرف.
5. طول معطية سلسلة أحرف.

[1] لمزيد من التفصيل حول العمليات على المعطيات من النوع سلسلة أحرف أنظر الفصل السادس

04.06.02 – العمليات على المتغيرات (المعطيات) من النوع حرف: [1]

تتمثل العمليات الممكن إجراءها على المعطيات من النوع سلسلة أحرف في ما يلي:

1. الدالة المعرفة مسبقا ord().
2. العمليات العلاقاتية.

05.06.02 – العمليات على المتغيرات (المعطيات) البولية :

تتمثل العمليات الممكن إجراءها على المتغيرات البولية في ما يلي:

1. الربط AND
2. الفصل OR
3. النفي NOT.

ملاحظة 04.02 :

يتم التصريح بالمتغيرات بجميع أنواعها باستخدام الكلمة المحجوزة Le Mot Clé Réservé ***var*** [2]

07.02 – الثوابت: Les Constantes

إضافة إلى المتغيرات و التي عبارة عن حيز من الذاكرة تشغل معطية معينة حيث أنه يمكن لهذا المحتوى أن يتغير خلال تنفيذ البرنامج، هناك ما يسمى بالثوابت Les Constantes

ملاحظة 05.02 :

يتم التصريح بالثوابت بجميع أنواعها باستخدام الكلمة المحجوزة Le Mot Clé Réservé ***const*** [3]

[1] لمزيد من التفصيل حول العمليات على المعطيات من النوع حرف أنظر الفصل السادس

[2] لمزيد من التفصيل حول التصريح بالمتغيرات أنظر الفصل الرابع

[3] لمزيد من التفصيل حول التصريح بالمتغيرات أنظر الفصل الرابع

تمارين الفصل

✍ التمـريـن 01.02:

1. ماذا نعني بالمعطيات.
2. ما هي أنواع المعطيات.

✍ التمـريـن 02.02:

1. ماذا نعني بالمعطيات الصحيحة.
2. أذكر مجموعة من المعطيات الصحيحة.

✍ التمـريـن 03.02:

1. ماذا نعني بالمعطيات الحقيقية.
2. أذكر مجموعة من المعطيات الحقيقية.

✍ التمـريـن 04.02:

1. ماذا نعني بالمعطيات سلسلة أحرف.
2. أذكر مجموعة من المعطيات سلسلة أحرف.

✍ التمـريـن 05.02:

1. ماذا نعني بالمعطيات من النوع حرف.
2. أذكر مجموعة من المعطيات من النوع حرف.

✍ التمـريـن 06.02:

لتكن المعطيات التالية:

-675	15000	123	2456
$234/:$	$+45?$	12.	130;456
$+768$	-15	-3456	9087
$+Z1709$	$+*RT45$	$+768.345$	-65
0.643	14500	$+467;009$	$+390$
-2409	$+546.780a$	E	$-\&IUF$

- **المطلـوب:**

1. حدد من بين المعطيات أعلاه تلك التي تعتبر معطيات صحيحة.

✍ التمـريـن 07.02:

لتكن المعطيات التالية:

• +	• 45.098	• 123;678	• 09,56
• 234/:	• +45?12,098	• −12.060	• 67.6*T*!
• +768.467	• −15.678	• −3456*TYH*	• 0.87
• 65*T*.34	• −56.98	• +768.345	• −65
• 0.643	• 14500	• +1,67	• 67.098
• −2409	• +546.780*a*	• *µ*£7.*UY*	• 345.098

- **المطلـوب:**

1. حدد من بين المعطيات أعلاه تلك التي تعتبر معطيات حقيقية.

✍ التمـريـن 08.02:

لتكن المعطيات التالية:

• −675	• 15000	• 123	• 2456
• 234/:	• +45?	• 12.	• 130;456
• +768	• −15	• −3456	• 9087
• +*Z*1709	• +**RT*45	• +768.345	• −65
• 0.643	• 14500	• +467;009	• +390
• −2409	• +546.780*a*	• *E*	• −&*IUF*

- **المطلـوب:**

1. حدد من بين المعطيات أعلاه تلك التي تعتبر معطيات سلسلة أحرف.

✍ التمـريـن 09.02:

لتكن المعطيات التالية:

• = *µ*	• 7	• 6*TYU*	• 2.*IKO*
• 67*RFD*	• 78§§	• 12.	• 48.*KJ*
• +768	• −15	• −3456	• *N*
• +*Z*1709	• /	• +768.345	• −65
• 0.643	• 14500	• +467;009	• +
• −2409	• !	• *X*	• −

- **المطلـوب:**

1. حدد من بين المعطيات أعلاه تلك التي تعتبر معطيات من النوع حرف.

✍ التمــريــن 10.02:

لتكن المعطيات التالية:

• $+NH\S/.$	• 869.75	• :/	• *NBK*?
• 234/:	• +45?	• 12.	• *TOLG*
• +768	• −15	• +67,87	• 90.87
• +Z1709	• 5	• +768.345	• −65
• 0.643	• 14500	• +467;009	• +390
• −09.009	• +6.78	• 8/76	• 7+7
• *true*	• *false*	• 4658	• 238,456

- **المطلـوب:**

1. حدد نوع كل معطية من بين المعطيات أعلاه.

✍ التمــريــن 11.02:

1. ماذا نعني بالمتغيرات.
2. كيف يسمى اسم المتغيرات .

✍ التمــريــن 12.02:

1. ما هي أنواع المتغيرات.
2. ما هو نوع المعطيات التي يتم تخزينها في كل نوع من المتغيرات.

✍ التمــريــن 13.02:

1. ما هي الكلمة المحجوزة Le Mot Clé Réservé المستخدمة للتصريح بالمتغيرات.
2. ما هي الكلمة المحجوزة Le Mot Clé Réservé المستخدمة للتصريح بالثوابت.

الفصـــل الثالـــث

التصريـــح بالمعطيـــات

Declaration Des Données

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- ماذا نعني بالتصريح بالمتغيرات.
- كيفية التصريح بالمتغيرات من النوع العددي
  1. كيفية التصريح بالمتغيرات العددية من النوع الصحيح.
  2. كيفية التصريح بالمتغيرات العددية من النوع الحقيقي.
- كيفية التصريح بالمتغيرات من النوع سلسلة أحرف.
- كيفية التصريح بالمتغيرات من النوع حرف.
- كيفية التصريح بالثوابت من النوع الصحيح.
- كيفية التصريح بالثوابت من النوع الحقيقي.
- كيفية التصريح بالثوابت من النوع سلسلة أحرف.

01.03 – تمهيـد:

أول عملية أو خطوة في صياغة و كتابة أي برنامج بلغة الـ PASCAL تتمثل في **التصريـح بالمعطيـات** (Declaration Des Données) الممثلة في المتغيرات و الثوابت . . . إلخ. عملية و خطوة التصريح بالمعطيات خاصة المتغيرات و الثوابت منها تعني تخصيص حيز أو خانة أو مكان من الذاكرة (Réservation d'un espace ou d'une case mémoire) لاستقبال و تخزين هذه المعطيات التي سوف يتم إدخالها إلى ذاكرة الحاسوب [1].

سوف يتم التطرق من خلال هذا الفصل إلى ما يلي:

1. Declaration des étiquettes ا
2. التصريح بالثوابت Declaration des constantes
3. التصريح بالأنواع Declaration des types
4. التصريح بالمتغيرات Declaration des variables
5. التصريح بالبرامج الفرعية و الدوال Declaration des procédures et fonctions

سوف يتم التطرق بالتفضيل إلى كيفية التصريح بكل من المتغيرات و الثوابت على أن يتم التطرق إلى كيفية التصريح بالعناصر الأخرى في الفصول اللاحقة.

02.03 – التصريـح بالمتغيـرات: Declaration des variables

نعني بعملية التصريح بالمتغيرات تخصيص حيز من الذاكرة (Un Espace Mémoire) أو تخصيص مكان في الذاكرة أو تخصيص خانة على مستوى الذاكرة (Case Mémoire) بغرض تخزين نوع معين من المعطيات من بين الأنواع العديدة من المعطيات [2]. سوف يتم التطرق من خلال هذه الفقرة إلى كيفية:

1. التصريح بالمتغيرات الصحيحة
2. التصريح بالمتغيرات الحقيقية
3. التصريح بالمتغيرات سلسلة أحرف
4. التصريح بالمتغيرات من النوع حرف

[1] لمزيد من التفصيل حول إدخال المعطيات أنظر الفصل الرابع

[2] لمزيد من التفصيل عن أنواع المعطيات أنظر الفصل الثاني

الجزء من البرنامج الذي يتم على مستواه أو فيه التصريح بالمتغيرات هو ذالك الجزء الواقع بين Le Mot Réservé ***Program*** و Le Mot Réservé ***begin*** كما هو موضح أدناه:

Program

الجزء

المخصص

للتصريح

بالمتغيرات

begin

يتم التصريح بالمتغيرات باستخدام الكلمة المحجوزة Le Mot Reservé ***Var*** وفقا لطريقة الكتابة التالية Suivant La Syntaxe Suivante:

var *Nom De La Variable* ***:*** *Type De La Variable* ***;***

حيث أن:

- ***Var*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالمتغيرات مهما كان نوعها (صحيحة، حقيقية، سلسلة أحرف أو من النوع حرف).
- *Nom De La Variable* : اسم المتغيرة التي نريد التصريح بها.
- *Type De La Variable* : يمثل نوع المتغيرة التي نريد التصريح بها حيث أن *Type De La Variable* تأخذ أحد Les Identificateurs Prédéfinis التالية:

1. ***Integer*** : و التي تعني أن المتغيرة المصرح بها من النوع الصحيح
2. ***Real*** : و التي تعني أن المتغيرة المصرح بها من النوع الحقيقي
3. ***String*** : و التي تعني أن المتغيرة المصرح بها من النوع سلسلة أحرف
4. ***Char*** : و التي تعني أن المتغيرة المصرح بها من النوع حرف

- ***;*** : فاصلة منقوطة Point Virgule

01.02.03 – التصريـح بالمتغيـرات الصحيحـة: Declaration des variables entières

يتم اللجوء إلى عملية التصريح بالمتغيرات الصحيحة إذا كان لدينا معطيات عددية صحيحة نريد إدخالها إلى الذاكرة بغرض تخزينها. تتم عملية التصريح بالمتغيرات العددية من النوع الصحيح (أي المتغيرات الصحيحة) وفقا بطريقة الكتابة التالية:

var *Nom De La Variable* ***: integer ;***

حيث أن:

- ***var*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالمتغيرة.
- *Nom De La Variable* : يمثل اسم المتغيرة الصحيحة التي نريد التصريح بها بغرض تخزين معطية صحيحة.
- ***:*** : تمثل نقطتين فوق بعضهما البعض.
- ***integer*** : يمثل Identificateur Prédéfinis يسمح بتحديد نوع المتغيرة التي تم التصريح بها بواسطة **Var** Le Mot Reservé. حيث أن ***integer*** تعني للحاسوب أن المتغيرة المصرح بها من النوع الصحيح.
- ***;*** : تمثل فاصلة منقوطة.

بغرض توضيح كيفية التصريح بالمتغيرات الصحيحة نأخذ المثال التالي:

مثـــال 01.03:

لتكن لدينا المعطية العددية من النوع الصحيح التالية:

4500

بغرض إدخال المعطية العددية من النوع الصحيح 4500 إلى ذاكرة الحاسوب و تخزينها بغرض استخدامها في العمليات الحسابية لاحقا يتم التصريح بمتغيرة عددية من النوع الصحيح تحمل الاسم *a* و التي على مستواها أو فيها يتم تخزين المعطية الصحيحة 4500. عملية التصريح تتم كما يلي:

var a : integer ;

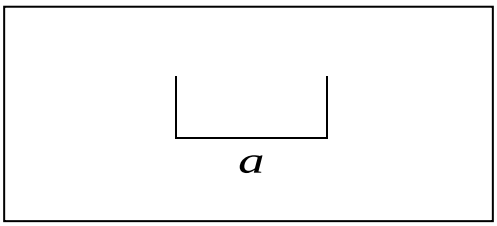
و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالمتغيرة من النوع الصحيح *a* هو التالي:

```
program DECLARATION ;
var a : integer ;
begin


end.
```

عند تنفيذ الحاسوب لهذا البرنامج فإنه يقوم بحجز أو بتخصيص حيز (Espace) أو مكان أو خانة (Case) على مستوى الذاكرة الحية (La RAM) يسميها *a* كما هو موضح أدناه:

الذاكرة الحيـة



ملاحظـة 01.03 :

مثلما يمكن التصريح بمتغيرة صحيحة واحدة من أجل إدخال و تخزين معطية صحيحة واحدة، يمكن التصريح بأكثر من متغيرة واحدة أي التصريح بالعديد من المتغيرات و هذا بغرض إدخال و تخزين العديد من المعطيات الصحيحة. تتم عملية التصريح بالعديد من المتغيرات العددية من النوع الصحيح وفقا لطريقة الكتابة التالية:

var *Nom_var_1* **,** *Nom_var_2* **,** *Nom_var_3* **,** *…….* **,** *Nom_var_n* ***: integer ;***

حيث أن:

- *Nom_var_1* , *Nom_var_2* , *Nom_var_3* , *……* , *Nom_var_n*: تمثل أسماء المتغيرات التي نريد التصريح بها. هذه المتغيرات تكون مفصولة بينها بفواصل.
- , : تمثل فاصلة تفصل أسماء المتغيرات المصرح بها.

بغرض التوضيح كيفية التصريح بالعديد من المتغيرات العددية من النوع الصحيح نأخذ المثال التالي:

مثـال 02.03:

لتكن لدينا المعطيات العددية الصحيحة التالية:

23 , 1250 , 150000 , 768 , 16800

بما أنه لدينا 5 معطيات عددية من النوع الصحيح نريد إدخالها إلى ذاكرة الحاسوب بغرض تخزينها، فإننا سوف نقوم بالتصريح بـ 5 متغيرات صحيحة تحمل الأسماء التالية:

a , b , c , d , e

تتم عملية التصريح كما يلي:

var *a* **,** *b* **,** *c* **,** *d* **,** *e* ***: integer ;***

نشير إلى أنه يمكن التصريح بالمتغيرات الصحيحة a , b , c , d , e كما يلي:

var *a* ***: integer ;***

```
var b : integer ;
var c : integer ;
var d : integer ;
var e : integer ;
```

كما يمكن كذالك التصريح بالمتغيرات الصحيحة a , b , c , d , e كما يلي:

```
var a : integer ;
    b : integer ;
    c : integer ;
    d : integer ;
    e : integer ;
```

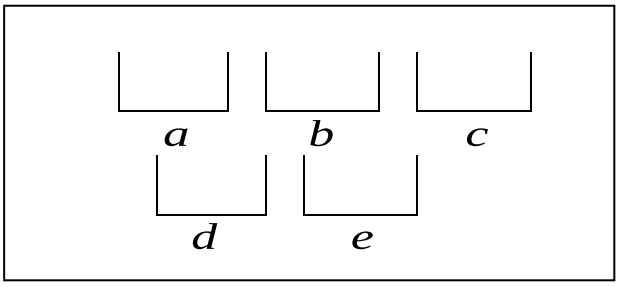
و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالمتغيرات من النوع الصحيح الصحيحة a , b , c , d , e هو التالي:

```
program DECLARATION ;
var a , b , c , d , e : integer ;
begin


end.
```

عند تنفيذ الحاسوب للبرنامج أعلاه فإنه يقوم بحجز أو بتخصيص 5 خانات على مستوى الذاكرة الحية (La RAM) نسميها a , b , c , d , e كما هو موضح أدناه:

الذاكرة الحيـة



02.02.03 – التصريـح بالمتغيـرات الحقيقيـة: Declaration des variables réelles

تتم عملية التصريح بالمتغيرات الحقيقية أو من النوع الحقيقي عندما نريد إدخال معطيات عددية حقيقية إلى الذاكرة بغرض تخزينها. تتم عملية التصريح بالمتغيرات من النوع الحقيقي (أي المتغيرات الحقيقية) وفقا لطريقة الكتابة التالية:

var *Nom De La Variable* ***: real ;***

حيث أن:

- ***var*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالمتغيرة.
- *Nom De La Variable* : يمثل اسم المتغيرة الصحيحة التي نريد التصريح بها بغرض تخزين معطية حقيقية.
- ***:*** : تمثل نقطتين فوق بعضهما البعض.
- ***real*:** يمثل Identificateur Prédéfinis يسمح بتحديد نوع المتغيرة التي تم التصريح بها بواسطة **Var** Le Mot Reservé. حيث أن ***real*** تعني للحاسوب أن المتغيرة المصرح بها من النوع الحقيقي.
- ***;*** : تمثل فاصلة منقوطة.

بغرض توضيح كيفية التصريح بالمتغيرات الصحيحة نأخذ المثال التالي:

مثـــال 03.03:

لتكن لدينا المعطية العددية من النوع الحقيقي التالية:

1250.345

بغرض إدخال المعطية الحقيقية 1250.345 إلى الذاكرة الحية للحاسوب و تخزينها بغرض استخدامها في العمليات الحسابية لاحقا يتم التصريح بمتغيرة عددية من النوع الحقيقي تحمل الاسم x و التي على محتواها أو فيها يتم تخزين المعطية من النوع الحقيقي 1250.345. عملية التصريح تتم كما يلي:

var x ***: real ;***

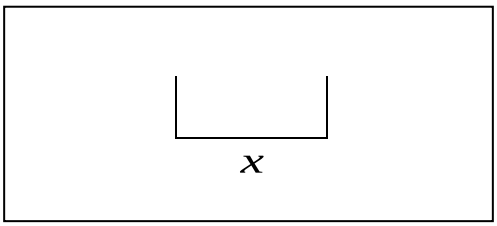
و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالمتغيرة من النوع الحقيقي x هو التالي:

```
program DECLARATION ;
var x : real ;
begin


end.
```

عند تنفيذ الحاسوب لهذا البرنامج فإنه يقوم بحجز و تخصيص خانة (Case) على مستوى الذاكرة الحية يسميها x كما هو موضح أدناه:

الذاكرة الحيـة



ملاحظـة 02.03 :

مثلما يمكن التصريح بمتغيرة حقيقية واحدة من أجل إدخال و تخزين معطية حقيقية واحدة، يمكن التصريح بأكثر من متغيرة حقيقية واحدة أي التصريح بالعديد من المتغيرات الحقيقية و هذا بغرض إدخال و تخزين العديد من المعطيات من النوع الحقيقي. تتم عملية التصريح بالعديد من المتغيرات العددية من النوع الحقيقي وفقا لطريقة الكتابة التالية:

var *Nom_var_1* **,** *Nom_var_2* **,** *Nom_var_3* **,** *…….* **,** *Nom_var_n* ***: real ;***

بغرض التوضيح كيفية التصريح بالعديد من المتغيرات العددية من النوع الحقيقي نأخذ المثال التالي:

مثـال 04.03:

لتكن لدينا المعطيات العددية من النوع الحقيقي التالية:

150.0789 , 0.450 , 15.00002 , 76.348 , 148.0901 , 123.0045

بما أنه لدينا 6 معطيات حقيقية نريد إدخالها إلى ذاكرة الحاسوب بغرض تخزينها، فإننا سوف نقوم بالتصريح بـ 6 متغيرات النوع الحقيقي تحمل الأسماء التالية:

x1 , x2 , x3 , x4 , x5 , x6

تتم عملية التصريح كما يلي:

var *x1 , x2 , x3 , x4 , x5 , x6* ***: real ;***

نشير إلى أنه يمكن التصريح بالمتغيرات الحقيقية x1 , x2 , x3 , x4 , x5 , x6 كما يلي:

var *x1* ***: real ;***

var *x2* ***: real ;***

var *x3* ***: real ;***

var *x4* ***: real ;***

var *x5* ***: real ;***

```
var x6 : real ;
```

كما يمكن كذالك التصريح بالمتغيرات الحقيقية x1 , x2 , x3 , x4 , x5 , x6 كما يلي:

```
var x1 : real ;
    x2 : real ;
    x3 : real ;
    x4 : real ;
    x5 : real ;
    x6 : real ;
```

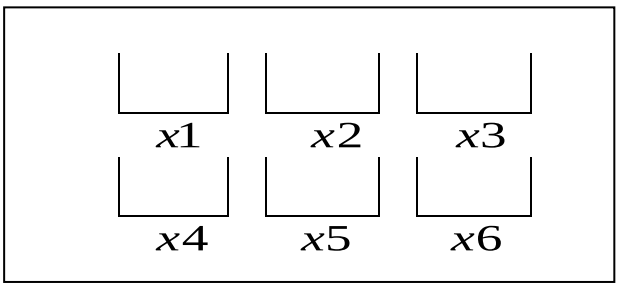
و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالمتغيرات من النوع الحقيقي , x1 , x2 x3 , x4 , x5 , x6 هو التالي:

```
program DECLARATION ;
var x1 , x2 , x3 , x4 , x5 , x6 : real ;
begin


end.
```

عند تنفيذ الحاسوب للبرنامج أعلاه فإنه يقوم بحجز و تخصيص 6 خانات على مستوى الذاكرة الحية يسميها x1 , x2 , x3 , x4 , x5 , x6 كما هو موضح أدناه:

الذاكرة الحيـة



03.02.03 – التصريـح بالمتغيـرات سلسلـة أحـرف: Declaration des variables chaine de caractères

يتم التصريح بالمتغيرات من النوع سلسلة أحرف وفقا لطريقة الكتابة التالية:

```
var Nom De La Variable : string [n] ;
```

حيث أن:

- ***var*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالمتغيرة.
- *Nom De La Variable* : يمثل اسم المتغيرة الصحيحة التي نريد التصريح بها بغرض تخزين معطية سلسلة أحرف.
- **:** : تمثل نقطتين فوق بعضهما البعض.
- ***string***: يمثل Identificateur Prédéfinis يسمح بتحديد نوع المتغيرة التي تم التصريح بها بواسطة Le Mot Reservé **Var**. حيث أن ***string*** تعني للحاسوب أن المتغيرة المصرح بها من النوع سلسلة أحرف.
- ***[ ]*** : Les Deux Crochets (gauche et droite)
- ***n*** : تمثل طول (عدد أحرف) المعطية سلسلة أحرف
- **;** : تمثل فاصلة منقوطة.

بغرض توضيح كيفية التصريح بالمتغيرات سلسلة أحرف نأخذ المثال التالي:

مثــال 05.03:

لتكن لدينا المعطية من النوع سلسلة أحرف التالية:

TIARET

بغرض إدخال المعطية سلسلة أحرف TIARET ذات الطول 6 حروف إلى ذاكرة الحاسوب و تخزينها بغرض استخدامها لاحقا في البرنامج يتم التصريح بمتغيرة من النوع سلسلة أحرف تحمل الاسم *w* و التي على مستواها أو فيها يتم تخزين المعطية من النوع سلسلة أحرف TIARET. عملية التصريح تتم كما يلي:

var w : string[6] ;

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالمتغيرة من النوع سلسلة أحرف *w* هو التالي:

```
program DECLARATION ;
var w : string[6] ;
begin


end.
```

ملاحظــة 03.03 :

مثلما يمكن التصريح بمتغيرة سلسلة أحرف واحدة من أجل إدخال و تخزين معطية سلسلة أحرف واحدة، يمكن التصريح بأكثر من متغيرة سلسلة أحرف واحدة أي التصريح بالعديد من

المتغيرات من النوع سلسلة أحرف و هذا بغرض إدخال و تخزين العديد من المعطيات من النوع سلسلة أحرف. تتم عملية التصريح بالعديد من المتغيرات سلسلة أحرف وفقا لطريقة الكتابة التالية:

var *Nom_var_1* **,** *Nom_var_2* **,** *Nom_var_3* **,** *……* **,** *Nom_var_n* **: *string[n] ;***

حيث أن:

بغرض التوضيح كيفية التصريح بالعديد من المتغيرات العددية من النوع سلسلة أحرف نأخذ المثال التالي:

مثـــال 06.03:

لتكن لدينا المعطيات من النوع سلسلة أحرف التالية:

Setti HAMID , ZAHRA , daira De SOUGUEUR , Freinda , tr=+148.0/ !9 , 345

بما أنه لدينا 6 معطيات سلسلة أحرف نريد إدخالها إلى ذاكرة الحاسوب بغرض تخزينها، فإننا سوف نقوم بالتصريح بـ 6 متغيرات من النوع سلسلة أحرف تحمل الأسماء التالية:

r1 , rr1 , a3 , b4 , xy , z

تتم عملية التصريح كما يلي:

var r1 , rr1 , a3 , b4 , xy , z ***: string ;***

نشير إلى أنه يمكن التصريح بالمتغيرات من النوع سلسلة أحرف r1 , rr1 , a3 , b4 , xy , z كما يلي:

var *r1* ***: string ;***

var *rr1* ***: string ;***

var *a3* ***: string ;***

var *b4* ***: string ;***

var *xy* ***: string ;***

var *z* ***: string ;***

كما يمكن كذالك التصريح بالمتغيرات من النوع سلسلة أحرف r1 , rr1 , a3 , b4 , xy , z كما يلي:

var r1 **: string ;**

rr1 **: string ;**

a3 **: string ;**

b4 **: string ;**

xy : **string ;**

z : **string ;**

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالمتغيرات من النوع سلسلة أحرف r1 , rr1 , a3 , b4 , xy , z هو التالي:

```
program DECLARATION ;
var  r1 , rr1 , a3 , b4 , xy , z  :  string ;
begin


end.
```

ملاحظـة 04.03 :

يمكن الاستغناء عن طول المعطية ***[n]*** عند التصريح بالمتغيرات من النوع سلسلة أحرف.

04.03.03 – التصريـح بالمتغيـرات من النـوع حـرف:Declaration des variables caractère

يتم التصريح بالمتغيرات من النوع حرف وفقا لطريقة الكتابة التالية:

```
var  Nom De La Variable  :  char ;
```

حيث أن:

- ***var*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالمتغيرة.
- *Nom De La Variable* : يمثل اسم المتغيرة الصحيحة التي نريد التصريح بها بغرض تخزين معطية عددية صحيحة.
- **:** : تمثل نقطتين فوق بعضهما البعض.
- ***char*** : يمثل Identificateur Prédéfinis يسمح بتحديد نوع المتغيرة التي تم التصريح بها بواسطة Le Mot Reservé **Var**. حيث أن ***char*** تعني للحاسوب أن المتغيرة المصرح بها من النوع حرف.
- **;** : تمثل فاصلة منقوطة.

بغرض توضيح كيفية التصريح بالمتغيرات من النوع حرف نأخذ المثال التالي:

مثـال 07.03:

لتكن لدينا المعطية من النوع حرف التالية:

R

بغرض إدخال المعطية من النوع حرف R إلى ذاكرة الحاسوب و تخزينها بغرض استخدامها لاحقا في البرنامج يتم التصريح بمتغيرة من النوع سلسلة أحرف تحمل الاسم *c* و التي على مستواها أو فيها يتم تخزين المعطية من النوع حرف R. عملية التصريح تتم كما يلي:

***var** c **: char** ;*

ملاحظـة 05.03 :

مثلما يمكن التصريح بمتغيرة واحدة من النوع حرف من أجل إدخال و تخزين معطية من النوع حرف واحدة، يمكن التصريح بأكثر من متغيرة واحدة أي التصريح بالعديد من المتغيرات من النوع حرف و هذا بغرض إدخال و تخزين العديد من المعطيات من النوع حرف. تتم عملية التصريح بالعديد من المتغيرات من النوع حرف وفقا لطريقة الكتابة التالية:

var** Nom_var_1 **,** Nom_var_2 **,** Nom_var_3 **, ……, ** Nom_var_n **: char ;

حيث أن:

بغرض التوضيح كيفية التصريح بالعديد من المتغيرات من النوع حرف نأخذ المثال التالي:

مثـال 08.03:

لتكن لدينا المعطيات من النوع حرف التالية:

S , Z , $, + , * , 3 , 5 , # , > , <

بما أنه لدينا 10 معطيات من النوع حرف نريد إدخالها إلى ذاكرة الحاسوب بغرض تخزينها، فإننا سوف نقوم بالتصريح بـ 10 متغيرات من النوع حرف تحمل الأسماء التالية:

x , x1 , y , yx , b , xy1 , z , i , y2 , y1

تتم عملية التصريح كما يلي:

var** x , x1 , y , yx , b , xy1 , z , i , y2 , y1 **: char ;

نشير إلى أنه يمكن التصريح بالمتغيرات من النوع حرف x , x1 , y , yx , b , xy1 , z , i , y2 , y1 كما يلي:

***var** x **: char** ;*

***var** x1 **: char** ;*

***var** y **: char** ;*

```
var yx : char ;
var b  : char ;
var xy1: char ;
var z  : char ;
var i  : char ;
var y2 : char ;
var y1 : char ;
```

كما يمكن كذالك التصريح بالمتغيرات من النوع حرف x , x1 , y , yx , b , xy1 , z , i , y2 , y1 كما يلي:

```
var x  : char ;
    x1 : char ;
    y  : char ;
    yx : char ;
    b  : char ;
    xy1: char ;
    z  : char ;
    i  : char ;
    y2 : char ;
    y1 : char ;
```

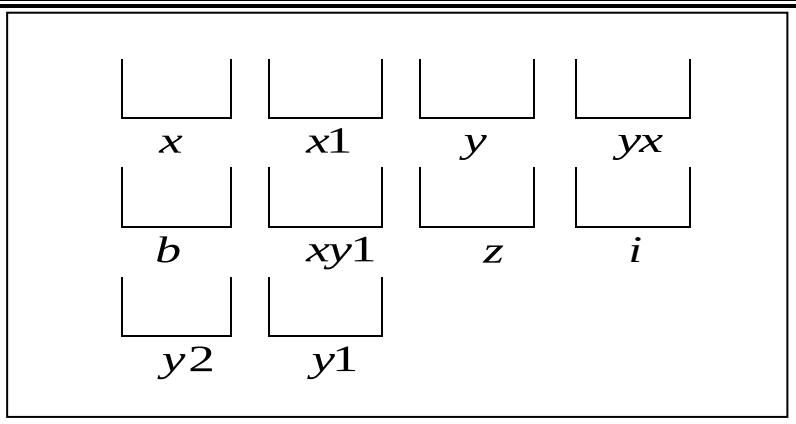
و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالمتغيرات من النوع حرف x , x1 , y , yx , b , xy1 , z , i , y2 , y1 هو التالي:

```
program DECLARATION ;
var x , x1 , y , yx , b , xy1 , z , i , y2 , y1 : char ;
begin


end.
```

عند تنفيذ الحاسوب لهذا البرنامج فإنه يقوم بحجز أو بتخصيص 10 خانات على مستوى الذاكرة الحية يسميها x , x1 , y , yx , b , xy1 , z , i , y2 , y1 كما هو موضح أدناه:

الذاكرة الحيـة



03.03 – التصريـح بالثوابـت: Declaration des constantes

مثلما يتم التصريح بالمتغيرات، فإنه بتم التصريح بالثوابت و ذلك باستخدام Le Mot Réservé ***Const*** وفقا لطريقة الكتابة التالية:

يتم التصريح بالمتغيرات من النوع حرف وفقا لطريقة الكتابة التالية:

const *Nom De La Constante = Donnée* ***;***

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثابتة La Constante.
- *Nom De La Constante* : يمثل اسم الثابتة (Identificateur) التي نريد التصريح بها.
- = : تمثل تساوي.
- *Donnée* : تمثل معطية أو قيمة الثابتة. هذه المعطية أو القيمة هي التي تحدد نوع الثابتة، حيث أنه:
  1. إذا كانت *Donnée* معطية صحيحة فإن الثابتة عبارة عن ثابتة صحيحة.
  2. إذا أخذت *Donnée* معطية حقيقية فإن الثابتة عبارة عن ثابتة حقيقية.
  3. إذا أخذت *Donnée* معطية سلسلة أحرف فإن الثابتة عبارة عن ثابتة سلسلة أحرف.
  4. إذا أخذت *Donnée* معطية من النوع حرف فإن الثابتة عبارة عن ثابتة من النوع حرف.
- **;** : تمثل فاصلة منقوطة.

سوف يتم التطرق من خلال هذه الفقرة إلى كيفية:

1. التصريح بالثوابت الصحيحة

2. التصريح بالثوابت الحقيقية
3. التصريح بالثوابت سلسلة أحرف
4. التصريح بالثوابت من النوع حرف

الجزء من البرنامج الذي يتم على مستواه أو فيه التصريح بالثوابت هو ذالك الجزء الواقع بين Le Mot Réservé ***Program*** و Le Mot Réservé ***begin*** كما هو موضح أدناه:

Program

الجزء

المخصص

للتصريح

بالمتغيرات

begin

01.03.03 – التصريـح بالثوابـت الصحيحـة: Declaration des constantes entières

يتم التصريح بالثوابت من النوع الصحيح وفقا لطريقة الكتابة التالية:

const *Nom De La Constante = Donnée_Entière* ***;***

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثابتة La Constante.
- *Nom De La Constante* : يمثل اسم الثابتة (Identificateur) التي نريد التصريح بها.
- = : تمثل تساوي.
- *Donnée_Entière*: تمثل **معطية صحيحة** التي تساويها الثابتة.

بغرض التوضيح كيفية التصريح بالثابتة العددية الصحيحة نأخذ المثال التالي:

مثـال 09.03:

لتكن لدينا المعطية العددية من النوع الصحيح التالية:

34509

يتم التصريح بهذه المعطية الصحيحة على أنها ثابتة عددية من النوع الصحيح تحمل الاسم *c* كما يلي:

const *c = 34509* ***;***

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالثابتة الصحيحة *c* هو التالي:

program *DECLARATION_DES_CONSTANTES ;*

```
const  c = 34509  ;
begin


end.
```

ملاحظـة 06.03 :

مثلما يمكن التصريح بثابتة عددية من النوع الصحيح واحدة، يمكن التصريح بأكثر من ثابتة عددية صحيحة أي التصريح بالعديد من الثوابت العددية من النوع الصحيح. تتم عملية التصريح بالعديد من الثوابت العددية الصحيحة وفقا لطريقة الكتابة التالية:

const NC1=DE1 **;** NC2=DE2 **;** NC2=DE2 **;** …… NCn=DEn **;**

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثوابت Les Constantes.
- NC1 : يمثل اسم الثابتة الأولى التي نريد التصريح بها.
- NC2 : يمثل اسم الثابتة الثانية التي نريد التصريح بها.
- NC3 : يمثل اسم الثابتة الثالثة التي نريد التصريح بها.
- ……
- NCn : يمثل اسم الثابتة رقم n التي نريد التصريح بها.
- = : تمثل تساوي.
- DE1 : تمثل معطية الثابتة الأولى التي نريد التصريح بها.
- DE2 : تمثل معطية الثابتة الثانية التي نريد التصريح بها.
- DE3 : تمثل معطية الثابتة الثالثة التي نريد التصريح بها.
- ……
- VEn : تمثل معطية الثابتة رقم n التي نريد التصريح بها.

بغرض التوضيح كيفية التصريح بالعديد من الثوابت العددية من النوع الصحيح نأخذ المثال التالي:

مثـال 10.03:

لتكن لدينا المعطيات العددية من النوع الصحيح التالية:

1250 , 4500 , 900 , 6789

يتم التصريح بهذه المعطيات الصحيحة على أنها ثوابت عددية من النوع الصحيح تحمل الأسماء , c1 c2 , c3 , c4 كما يلي:

```
const  c1 = 1250  ; c2 = 4500  ;  c3 = 900  ;  c4 = 6789  ;
```

نشير إلى أنه يمكن التصريح بالثوابت من النوع الصحيح c1 , c2 , c3 , c4 كما يلي:

```
const  c1 = 1250  ;
const  c2 = 4500  ;
const  c3 = 900   ;
const  c4 = 6789  ;
```

كما يمكن كذالك التصريح بالثوابت من النوع الصحيح c1 , c2 , c3 , c4 كما يلي:

```
const  c1 = 1250  ;
       c2 = 4500  ;
       c3 = 900   ;
       c4 = 6789  ;
```

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالثوابت الصحيحة c1 , c2 , c3 , c4 هو التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  c1 = 1250 ;
       c2 = 4500  ;
       c3 = 900   ;
       c4 = 6789 ;
begin


end.
```

02.03.03 – التصريـح بالثوابـت الحقيقيـة: Declaration des constantes réelles

يتم التصريح بالثوابت من النوع الحقيقي وفقا لطريقة الكتابة التالية:

```
const  Nom De La Constante  = Donnée_Réelle  ;
```

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثابتة La Constante.

- *Nom De La Constante* : يمثل اسم الثابتة التي نريد التصريح بها.
- = : تمثل تساوي.
- *donnée_ Réelle*: تمثل **معطية حقيقية** التي تساويها الثابتة.

بغرض التوضيح كيفية التصريح بالثابتة العددية الحقيقية نأخذ المثال التالي:

مثـال 11.03:

لتكن لدينا المعطية العددية من النوع الصحيح التالية:

1240.567

يتم التصريح بهذه المعطية الصحيحة على أنها ثابتة عددية من النوع الحقيقي تحمل الاسم *c* كما يلي:

const *c* = 1240.567 ***;***

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالثابتة الصحيحة *c* هو التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  c = 1240.567  ;
begin


end.
```

ملاحظـة 07.03 :

مثلما يمكن التصريح بثابتة عددية من النوع الحقيقي واحدة، يمكن التصريح بأكثر من ثابتة عددية حقيقية أي التصريح بالعديد من الثوابت العددية من النوع الحقيقي. تتم عملية التصريح بالعديد من الثوابت العددية الحقيقية وفقا لطريقة الكتابة التالية:

const NC1=DR1 **;** NC2=DR2 **;** NC2=DR2 **;** …… NCn=DRn **;**

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثوابت Les Constantes.
- NC1 : يمثل اسم الثابتة الأولى التي نريد التصريح بها.
- NC2 : يمثل اسم الثابتة الثانية التي نريد التصريح بها.
- NC3 : يمثل اسم الثابتة الثالثة التي نريد التصريح بها.
- ……

- NCn : يمثل اسم الثابتة رقم n التي نريد التصريح بها.
- = : تمثل تساوي.
- DR1 : تمثل معطية الثابتة الأولى التي نريد التصريح بها.
- DR2 : تمثل معطية الثابتة الثانية التي نريد التصريح بها.
- DR3 : تمثل معطية الثابتة الثالثة التي نريد التصريح بها.
- .....
- DRn : تمثل معطية الثابتة رقم n التي نريد التصريح بها.

بغرض التوضيح كيفية التصريح بالعديد من الثوابت العددية من النوع الحقيقي نأخذ المثال التالي:

مثـال 12.03:

لتكن لدينا المعطيات العددية من النوع الحقيقي التالية:

12.50 , 0.4500 , 900.123 , 67.8904

يتم التصريح بهذه المعطيات الصحيحة على أنها ثوابت عددية من النوع الحقيقي تحمل الأسماء c1 , c2 , c3 , c4 كما يلي:

```
const  c1 = 12.50 ; c2 = 0.4500 ;  c3 = 900.123 ;  c4 = 67.8904 ;
```

نشير إلى أنه يمكن التصريح بالثوابت من النوع الحقيقي c1 , c2 , c3 , c4 كما يلي:

```
const  c1 = 12.50 ;
const  c2 = 0.4500 ;
const  c3 = 900.123  ;
const  c4 = 67.8904 ;
```

كما يمكن كذالك التصريح بالثوابت من النوع الحقيقي c1 , c2 , c3 , c4 كما يلي:

```
const  c1 = 12.50 ;
       c2 = 0.4500 ;
       c3 = 900.123  ;
       c4 = 67.8904 ;
```

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالثوابت الحقيقية c1 , c2 , c3 , c4 هو التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  c1 = 12.50 ;
       c2 = 0.4500 ;
```

```
c3 = 900.123  ;
c4 = 67.8904 ;
begin

end.
```

03.03.03 – التصريـح بالثوابـت سلسلـة أحـرف: Declaration des constantes chaine de caractères

يتم التصريح بالثوابت من النوع سلسلة أحرف وفقا لطريقة الكتابة التالية:

```
const  Nom De La Constante  = ' Donnée_Chaine_De_Caractère '  ;
```

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثابتة La Constante.
- *Nom De La Constante* : يمثل اسم الثابتة التي نريد التصريح بها.
- = : تمثل تساوي.
- *Donnée_Chaine_De_Caractère*: تمثل **معطية من النوع سلسلة أحرف** التي تساويها الثابتة.
- Deux Qotes : ' '

بغرض التوضيح كيفية التصريح بالثابتة سلسلة أحرف نأخذ المثال التالي:

مثـال 13.03:

لتكن لدينا المعطية العددية من النوع السلسلة أحرف التالية:

Sougueur

يتم التصريح بهذه المعطية سلسلة أحرف على أنها ثابتة من النوع سلسلة أحرف تحمل الاسم *ch* كما يلي:

```
const  ch = 'Sougueur' ;
```

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالثابتة سلسلة أحرف *ch* هو التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  ch = 'Sougueur' ;
begin
```

end.

ملاحظـة 08.03 :

مثلما يمكن التصريح بثابتة من النوع سلسلة أحرف واحدة، يمكن التصريح بأكثر من ثابتة سلسلة أحرف أي التصريح بالعديد من الثوابت سلسلة أحرف. تتم عملية التصريح بالعديد من الثوابت من النوع سلسلة أحرف وفقا لطريقة الكتابة التالية:

const NC1='DC1' **;** NC2='DC2' **;** NC2='DC2' **;** …… NCn='DCn' **;**

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثوابت Les Constantes.
- NC1 : يمثل اسم الثابتة الأولى التي نريد التصريح بها.
- NC2 : يمثل اسم الثابتة الثانية التي نريد التصريح بها.
- NC3 : يمثل اسم الثابتة الثالثة التي نريد التصريح بها.
- ……
- NCn : يمثل اسم الثابتة رقم n التي نريد التصريح بها.
- = : تمثل تساوي.
- DC1 : تمثل معطية الثابتة الأولى التي نريد التصريح بها.
- DC2 : تمثل معطية الثابتة الثانية التي نريد التصريح بها.
- DC3 : تمثل معطية الثابتة الثالثة التي نريد التصريح بها.
- ……
- DCn : تمثل معطية الثابتة رقم n التي نريد التصريح بها.

بغرض التوضيح كيفية التصريح بالعديد من الثوابت من النوع سلسلة أحرف نأخذ المثال التالي:

مثـال 14.03:

لتكن لدينا المعطيات العددية من النوع سلسلة أحرف التالية:

Tiaret14 , SIHEM , ?/§+=< 900.123 , Infor , 567890

يتم التصريح بهذه المعطيات من النوع سلسلة أحرف على أنها ثوابت سلسلة أحرف تحمل الأسماء c1 , c2 , c3 , c4 , c5 كما يلي:

const *c1* = ‘Tiaret14’***;*** *c2* = ‘SIHEM’***;*** *c3* = ‘?/§+=< 900.123’***;*** *c4* = ‘Infor’***;*** *c5* = ‘567890’***;***

نشير إلى أنه يمكن التصريح بالثوابت من النوع سلسلة أحرف c1 , c2 , c3 , c4 ,c5 كما يلي:

const *c1* = ‘Tiaret14’ ***;***

const *c2* = ‘SIHEM’***;***

const *c3* = ‘?/§+=< 900.123’ ***;***

const *c4* = ‘Infor’***;***

const *c5* = ‘567890’***;***

كما يمكن كذالك التصريح بالثوابت من النوع سلسلة أحرف c1 , c2 , c3 , c4 ,c5 كما يلي:

const c1 = ‘Tiaret14’ **;**

c2 = ‘SIHEM’ **;**

c3 = ‘?/§+=< 900.123’ **;**

c4 = ‘Infor’ **;**

c5 = ‘567890’ **;**

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالثوابت سلسلة أحرف c1 , c2 , c3 , c4 ,c5 هو التالي:

program *DECLARATION_DES_CONSTANTES ;*

const c1 = ‘Tiaret14’ **;**

c2 = ‘SIHEM’ **;**

c3 = ‘?/§+=< 900.123’ **;**

c4 = ‘Infor’ **;**

c5 = ‘567890’ **;**

begin

end.

04.03.03 – التصريـح بالثوابـت من النوع حـرف: Declaration des constantes caractère

يتم التصريح بالثوابت من النوع حرف وفقا لطريقة الكتابة التالية:

const *Nom De La Constante = ‘ Donnée_De_Type_Caractère ‘ ;*

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثابتة La Constante.
- *Nom De La Constante* : يمثل اسم الثابتة التي نريد التصريح بها.
- = : تمثل تساوي.
- *Donnée_De_Type_Caractère*: تمثل **معطية من النوع حرف** التي تساويها الثابتة.
- Deux quotes : ‘ ‘

بغرض التوضيح كيفية التصريح بالثابتة من النوع حرف نأخذ المثال التالي:

مثـال 15.03:

لتكن لدينا المعطية العددية من النوع السلسلة أحرف التالية:

Z

يتم التصريح بهذه المعطية من النوع حرف على أنها ثابتة من النوع حرف تحمل الاسم *ch* كما يلي:

const *ch = ‘Z’* ***;***

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالثابتة من النوع حرف *ch* هو التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  ch = ‘Z’ ;
begin


end.
```

ملاحظـة 09.03 :

مثلما يمكن التصريح بثابتة من النوع حرف واحدة، يمكن التصريح بأكثر من ثابتة أي التصريح بالعديد من الثوابت من النوع حرف. تتم عملية التصريح بالعديد من الثوابت من النوع حرف وفقا لطريقة الكتابة التالية:

const NC1=’DC1’ **;** NC2=’DC2’ **;** NC2=’DC2’ **;** …… NCn=’DCn’ **;**

حيث أن:

- ***const*** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالثوابت Les Constantes.
- NC1 : يمثل اسم الثابتة الأولى التي نريد التصريح بها.

- NC2 : يمثل اسم الثابتة الثانية التي نريد التصريح بها.
- NC3 : يمثل اسم الثابتة الثالثة التي نريد التصريح بها.
- .....
- NCn : يمثل اسم الثابتة رقم n التي نريد التصريح بها.
- = : تمثل تساوي.
- DC1 : تمثل معطية الثابتة الأولى التي نريد التصريح بها.
- DC2 : تمثل معطية الثابتة الثانية التي نريد التصريح بها.
- DC3 : تمثل معطية الثابتة الثالثة التي نريد التصريح بها.
- .....
- DCn : تمثل معطية الثابتة رقم n التي نريد التصريح بها.

بغرض التوضيح كيفية التصريح بالعديد من الثوابت من النوع سلسلة أحرف نأخذ المثال التالي:

مثـال 16.03:

لتكن لدينا المعطيات العددية من النوع حرف التالية:

T , % , ? , / , > , ! , 9 , { , [, ^

يتم التصريح بهذه المعطيات من النوع سلسلة أحرف على أنها ثوابت تحمل الأسماء r1 , r2 , r3 , r4 , r5 , r6 , r7 , r8 , r9 , r10 كما يلي:

const *r1 = 'T' ; r2 = '%' ; r3 = '?' ; r4 = '/' ; r5 = '>' ; r6 = '9' ; r7 = '{' ;*

نشير إلى أنه يمكن التصريح بالثوابت من النوع حرف r1 , r2 , r3 , r4 , r5 , r6 , r7 , r8 , r9 , r10 كما يلي:

const *r1 = 'T' ;*

const *r2 = '%' ;*

const *r3 = '?' ;*

const *r4 = '/' ;*

const *r5 = '>' ;*

const *r6 = '9' ;*

const *r7 = '{' ;*

كما يمكن كذالك التصريح بالثوابت من النوع سلسلة أحرف r1 , r2 , r3 , r4 , r5 , r6 , r7 , r8 , r9 , r10 , كما يلي:

const r1 = 'T' ;

r2 = ‘%’ ;

r3 = ‘?’ ;

r4 = ‘/’ ;

r5 = ‘>’ ;

r6 = ‘9’ ;

r7 = ‘{’ ;

و عليه فإن البرنامج بلغة الـ PASCAL الذي يسمح بالتصريح بالثوابت سلسلة أحرف r1 , r2 , r3 , r4 , r5 , r6 , r7 , r8 , r9 , r10 هو التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  r1 = ‘T’  ;
       r2 = ‘%’ ;
       r3 = ‘?’  ;
       r4 = ‘/’ ;
       r5 = ‘>’ ;
       r6 = ‘9’ ;
       r7 = ‘{’ ;
begin

end.
```

تمـاريـــن الفصـــــل

✍ التمــريــن 01.03:

1. قدم Le Mot Réservé الذي يسمح بالتصريح بالمتغيرات (Declaration Des Variables).

✍ التمــريــن 02.03:

1. قدم طريقة الكتابة (La Syntaxe) التي تسمح بالتصريح بمتغيرة من النوع **الصحيح** (Une Variable De Type Entier).

2. قدم مثالا على طريقة الكتابة المتوصل إليها في السؤال الأول.

✍ التمــريــن 03.03:

1. أكتب البرنامج الذي يسمح بالتصريح بمتغيرة من النوع **الصحيح** و لتكن تحمل الاسم *a*

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول

✍ التمــريــن 04.03:

1. أكتب البرنامج الذي يسمح بالتصريح بأربع متغيرات من النوع **الصحيح** و لتكن تحمل الأسماء التالية: *b*، *c*2، *d*، *a*1.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 05.03:

لتكن المعطيات من النوع الصحيح التالية:

1245 , 45890 , 56790 , 150000

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 06.03:

ليكن البرنامج التالي:

```
program XXXX ;
var  alg , gvf4 , c9rt : integer ;
```

```
var  x1, z5u3 , val , t3ab : integer ;
var  s : integer ;
begin

end.
```

- **المطلـوب:**

1. حدد ماذا يقوم البرنامج أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج أعلاه.

✍ التمــريــن 07.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  g , g4 , rt : integer ;
var  i : integer ;
var  d : integer ;
begin

end.
```

- **المطلـوب:**

1. حدد ماذا يقوم البرنامج أعلاه.

2. أعد كتابة البرنامج بطريقة أخرى.

3. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج أعلاه.

✍ التمــريــن 08.03:

لتكن المعطيات الصحيحة التالية:

3589 , 3450 , 8900

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

التمـريــن 09.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  a  b  c : integer ;
var  r : integer ;
var  y : integer ;
begin

end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

التمـريــن 10.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  x , y , z : integer ;
var  w  integer ;
var  t : integer
begin

end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

التمـريــن 11.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
```

```
var  zx , zy , zz : integer ;
var  z  integer ;
     v : integer ;
begin


end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 12.03:

1. قدم طريقة الكتابة (La Syntaxe) التي تسمح بالتصريح (Declaration) بمتغيرة من النوع **الحقيقي** (Une Variable De Type Réel).

2. قدم مثالا على طريقة الكتابة المتوصل إليها في السؤال الأول.

✍ التمـريـن 13.03:

1. أكتب البرنامج الذي يسمح بالتصريح بمتغيرة من النوع **الحقيقي** و لتكن تحمل الاسم *r*

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول

✍ التمـريـن 14.03:

1. أكتب البرنامج الذي يسمح بالتصريح بأربع متغيرات من النوع **الحقيقي** و لتكن تحمل الأسماء التالية: *t*، *nombre*، *x*1، *azr*1

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول

✍ التمـريـن 15.03:

ليكن البرنامج التالي:

```
program YYYY ;
var  fr , va , x : real ;
var  x1, ur , k1 : real ;
begin
```

end.

- **المطلـوب:**

1. حدد ماذا يقوم البرنامج أعلاه.

2. أعد كتابة البرنامج بطريقة أخرى.

3. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج أعلاه.

✍ التمـريـن 16.03:

ليكن البرنامج التالي:

```
program ZERTTTT ;
var  dfm1 , valeur , xy7 : real ;
var  j1, valeur1 , j1k1 , j5k2 : real ;
var  w3xy : real ;
var  valeur3 : real ;
begin


end.
```

- **المطلـوب:**

1. حدد ماذا يقوم البرنامج أعلاه.

2. أعد كتابة البرنامج بطريقة أخرى.

3. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج أعلاه.

✍ التمـريـن 17.03:

لتكن المعطيات العددية من النوع الحقيقي التالية:

1250.349 , 0.001 , 500.125 , 670.89

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمـريـن 18.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  a  b  c : real ;
var  r : real ;
var  y : real ;
begin

end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 19.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  x  ,  y  ,  z : real ;
var  w  real ;
var  t : real
begin

end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 20.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  zx  ,  zy  ,  zz : real ;
var  z  real ;
```

v ***: real ;***

begin

end.

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 21.03:

1. قدم طريقة الكتابة (La Syntaxe) التي تسمح بالتصريح بمتغيرة من النوع **سلسلة أحرف** (Une Variable De Type Chaine De Caractères).

2. قدم مثالا على طريقة الكتابة المتوصل إليها في السؤال الأول.

✍ التمـريـن 22.03:

1. أكتب البرنامج الذي يسمح بالتصريح بمتغيرة من النوع **سلسلة أحرف** و لتكن تحمل الاسم *f*

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول

✍ التمـريـن 23.03:

1. أكتب البرنامج الذي يسمح بالتصريح بأربع متغيرات من النوع **سلسلة أحرف** و لتكن تحمل الأسماء التالية: *no*، *adr*، *nom*، *nom*1

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول

✍ التمـريـن 24.03:

لتكن المعطيات من النوع سلسلة أحرف التالية:

La Wilaya De Tiaret , Zahra , FREIDA , VOL0.125 , N+= !:§%ù

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

.3

التمــريــن 25.03:

ليكن البرنامج التالي:

```
program YYY ;
var  dimr1 , t3r , xy7 : string  ;
var  j1, j3 , jk1 , jk2 : string ;
begin


end.
```

- **المطلـوب:**

1. بماذا يقوم البرنامج أعلاه.

2. أعد كتابة البرنامج بطريقة أخرى.

التمــريــن 26.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  zx  ,  zy  ,  zz : string
var  z  string ;
begin


end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

التمــريــن 27.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  a  ,  b  ,  c  : string
var  h  string
begin
```

end.

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 28.03:

ليكن البرنامج التالي:

```
program DECLARATION ;
var  ch1  , ch2   c  : string ;
var  h  : string
begin


end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 29.03:

لتكن المعطيات من النوع سلسلة أحرف التالية:

SIHEM , &é »HYT , 123467 , REGINA , SOUGUEUR

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمـريـن 30.03:

1. قدم طريقة الكتابة (La Syntaxe) التي تسمح بالتصريح (Declaration) بمتغيرة من النوع **حرف** (Une Variable De Type caractère).

2. قدم مثالا على طريقة الكتابة المتوصل إليها في السؤال الأول.

✍ التمــريــن 31.03:

1. أكتب البرنامج الذي يسمح بالتصريح بمتغيرة من النوع **حرف** و لتكن تحمل الاسم *w* .

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 32.03:

1. أكتب البرنامج الذي يسمح بالتصريح بأربع متغيرات من النوع **حرف** و لتكن تحمل الأسماء التالية: *car*، *a*، *oui*، *oui*1 .

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 33.03:

لتكن المعطيات من النوع حرف التالية:

L , / , ? , à , * , = , ° , 7 , 9

- **المطلــوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 34.03:

ليكن البرنامج التالي:

```
program ZZZ ;
var  dtr1 , y11 , z56x : char ;
var  rtj1, ytj1 , ujk6 , jk : char ;
begin

end.
```

- **المطلــوب:**

1. بماذا يقوم البرنامج أعلاه.

2. أعد كتابة البرنامج بطريقة أخرى.

✍ التمــريــن 35.03:

ليكن البرنامج التالي:

```
program DECLA ;
var  d , y , z5  char ;
var  r  u  , k1 , j7  : char ;
var  t , k10  : har
begin


end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 36.03:

ليكن البرنامج التالي:

```
program ttttt ;
var  t1 , t2 , t5 : char ;
var  rt  , u8  ,  : char ;
begin


end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 37.03:

ليكن البرنامج التالي:

```
program YYY ;
var  ru , ngh2t , qop4 , p : integer ;
var  val , val3 , b , klm6jk2 : real ;
var  nom11 , c , y5 : char ;
var  noqm , t , pprenom : string ;
```

```
var  r , n , qop5 , p3 : integer ;
var  nom112 , c2 , y53 : char ;
var  norqm , tr , pprenomr : string ;
var  val5 , val33 , br , klm6jk2t : real ;
begin


end.
```

- **المطلـوب:**

1. بماذا يقوم البرنامج أعلاه.

✍ التمـريـن 38.03:

ليكن البرنامج التالي:

```
program declaration ;
var  r , n , q  p : integer ;
var  v , v3 , k , k2  real ;
var  n1 , c0 , no : char
var  oq , t , p : strig[3] ;
begin


end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 39.03:

لتكن لدينا المعطيات العددية التالية:

23 , 1250 , 150000 , 768 , 16800

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 40.03:

لتكن لدينا المعطيات العددية التالية:

150.0789 , 0.450 , 15.00002 , 76.348 , 148.0901 , 123.0045

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 41.03:

لتكن لدينا المعطيات التالية:

Setti HAMID , ZAHRA , daira De SOUGUEUR , Freinda , tr=+148.0/ !9 , 345

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

✍ التمــريــن 42.03:

لتكن لدينا المعطيات التالية:

S , Z , $, + , * , 3 , 5 , # , > , <

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالمتغيرات التي يتم فيها تخزين المعطيات أعلاه.

2. وضح ماذا يحدث على مستوى الذاكرة الحية عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 43.03:

1. قدم Le Mot Réservé الذي يسمح بالتصريح بالثوابت (Declaration Des Constantes).

✍ التمــريــن 44.03:

1. قدم طريقة الكتابة (La Syntaxe) التي تسمح بالتصريح بثابتة من النوع **الصحيح** (Une Constante De Type Entier).

2. قدم مثالا على طريقة الكتابة المتوصل إليها في السؤال الأول.

✍ التمــريــن 45.03:

1. أكتب البرنامج الذي يسمح بالتصريح بثابتة من النوع **الصحيح** و لتكن تحمل الاسم *a*

✍ التمــريــن 46.03:

1. أكتب البرنامج الذي يسمح بالتصريح بأربع ثوابت من النوع **الصحيح** و لتكن تحمل الأسماء التالية: *b*، *c2*، *d*، *a1*.

✍ التمــريــن 47.03:

لتكن المعطيات من النوع الصحيح التالية:

1245 , 45890 , 56790 , 150000

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

✍ التمــريــن 48.03:

لتكن المعطيات التالية:

1250 , 4500 , 900 , 6789

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

✍ التمــريــن 49.03:

ليكن البرنامج التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  c1 = 34578 ;
       c2 = 123  ;
       c3 = 7890   ;
       c4 = 9807 ;
begin


end.
```

- **المطلـوب:**

1. بماذا يقوم البرنامج أعلاه.

✍ التمــريــن 50.03:

ليكن البرنامج التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  re1 = 2345
       yu : = 700  ;
       c3 : 678   ;
       ct = 789 ;
begin


end.
```

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمــريــن 51.03:

1. قدم طريقة الكتابة (La Syntaxe) التي تسمح بالتصريح بثابتة من النوع **الحقيقي** (Une Constante De Type Réel).

2. قدم مثالا على طريقة الكتابة المتوصل إليها في السؤال الأول.

✍ التمــريــن 52.03:

1. أكتب البرنامج الذي يسمح بالتصريح بثابتة من النوع **الحقيقي** و لتكن تحمل الاسم *r*

✍ التمــريــن 53.03:

1. أكتب البرنامج الذي يسمح بالتصريح بأربع ثوابت من النوع **الحقيقي** و لتكن تحمل الأسماء التالية: *t*، *nombre*، *x*1، *azr*1

✍ التمــريــن 54.03:

لتكن المعطيات التالية:

1250.349 , 0.001 , 500.125 , 670.89

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

✍ التمـريـن 55.03:

لتكن المعطيات التالية:

12.50 , 0.4500 , 900.123 , 67.8904

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

✍ التمـريـن 56.03:

ليكن البرنامج التالي:

```
program DECLARATION_DES_CONSTANTES ;
const  c1 = 670.87 ;
const  c2 = 34.890  ;
const  c3 = 0.2367   ;
const  c4 = 79.0001 ;
begin


end.
```

- **المطلـوب:**

4. حدد ماذا يقوم البرنامج أعلاه.

5. أعد كتابة البرنامج بطريقة أخرى.

✍ التمـريـن 57.03:

ليكن البرنامج التالي:

```
program ZERTTTT ;
const  c1 := 0.87 ;
const  c2 = 4.009
const  c3 = 8.299   ;
const  c4 = 9.056 ;
```

begin

end.

- **المطلـوب:**

1. حدد فيما إذا كان البرنامج أعلاه يحتوي على أخطاء.

2. قدم البرنامج أعلاه بدون أخطاء.

✍ التمـريـن 58.03:

1. قدم طريقة الكتابة (La Syntaxe) التي تسمح بالتصريح بثابتة من النوع **سلسلة أحرف** (Une Constante De Type Chaine De Caractères).

2. قدم مثالا على طريقة الكتابة المتوصل إليها في السؤال الأول.

✍ التمـريـن 59.03:

1. أكتب البرنامج الذي يسمح بالتصريح بثابتة من النوع **سلسلة أحرف** و لتكن تحمل الاسم *f*

✍ التمـريـن 60.03:

3. أكتب البرنامج الذي يسمح بالتصريح بأربع ثوابت من النوع **سلسلة أحرف** و لتكن تحمل الأسماء التالية: *no*، *adr*، *nom*، *nom*1

✍ التمـريـن 61.03:

لتكن المعطيات التالية:

La Wilaya De Tiaret , Zahra , FREIDA , VOL0.125 , N+= !:§%ù

- **المطلـوب:**

1. ما هو نوع المعطيات أعلاه.

2. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

✍ التمـريـن 62.03:

لتكن المعطيات من النوع سلسلة أحرف التالية:

Tiaret14 , SIHEM , ?/§+=< 900.123 , Infor , 567890

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

✍ التمــريــن 63.03:

لتكن المعطيات من النوع سلسلة أحرف التالية:

volani , &é »HYT , 123467 , REGINA , sonatiba

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

✍ التمــريــن 64.03:

1. قدم طريقة الكتابة التي تسمح بالتصريح بثابتة من النوع **حرف** (Une Constante De Type caractère).

2. قدم مثالا على طريقة الكتابة المتوصل إليها في السؤال الأول.

✍ التمــريــن 65.03:

1. أكتب البرنامج الذي يسمح بالتصريح بثابتة من النوع **حرف** و لتكن تحمل الاسم *w* .

✍ التمــريــن 66.03:

1. أكتب البرنامج الذي يسمح بالتصريح بأربع ثوابت من النوع **حرف** و لتكن تحمل الأسماء التالية:

*car، a، oui، oui*1 .

✍ التمــريــن 67.03:

لتكن المعطيات من النوع حرف التالية:

L , / , ? , à , * , = , ° , 7 , 9

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

✍ التمــريــن 68.03:

لتكن المعطيات من النوع حرف التالية:

T , % , ? , / , > , ! , 9 , { , [, ^

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالتصريح بالثوابت التي يتم فيها تخزين المعطيات أعلاه.

الفصـــل 🕮 الرابـــع

ادخـــــال المعطيـــــات

Introduction Des Données

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- إدخال المعطيات عن طريق البرنامج.
  1. إدخال المعطيات العددية من النوع الصحيح.
  2. إدخال المعطيات العددية من النوع الحقيقي.
  3. إدخال المعطيات من النوع سلسلة حروف.
  4. إدخال المعطيات من النوع حرف.

- إدخال المعطيات عن طريق لوحة المفاتيح.
  5. إدخال المعطيات العددية من النوع الصحيح.
  6. إدخال المعطيات العددية من النوع الحقيقي.
  7. إدخال المعطيات من النوع سلسلة حروف.
  8. إدخال المعطيات من النوع حرف.

01.04 – تمهيـد:

بعد العملية أو الخطوة الأولى الممثلة في التصريح بالمعطيات (الممثلة في المتغيرات و الثوابت) اللازمة و التي يحتاجها البرنامج نقوم بالخطوة الثانية و الممثلة في عملية **إدخـال المعطيـات** (Introduction Des Données). هذه العملية أي إدخال المعطيات تتم في لغة الـ PASCAL وفقا أو بطريقتين هما:

1. إدخال المعطيات عن طريق البرنامج Introduction des données par le programme
2. إدخال المعطيات عن طريق لوحة المفاتيح Introduction des données par le clavier

سوف يتم التطرق بالتفضيل إلى كيفية إدخال المعطيات وفقا لكل طريقة مذكورة أعلاه.

02.04 – إدخـال المعطيـات عن طريـق البرنامـج: التوجيـه المباشـر

Introduction des données par le programme : L'affectation directe

تبعا لهذه الطريقة لإدخال المعطيات، فإن معطيات المتغيرات (مهما كان نوع المتغيرة) تكون موجودة في أو داخل البرنامج. يتم إدخال المعطيات تبعا لهذه الطريقة وفقا لطريقة الكتابة التالية Suivant La Syntaxe Suivante:

Variable **:=** *Donnée* **;**

حيث أن:

- *Variable* : تمثل اسم المتغيرة التي نريد أن نوجه لها أو نخزن فيها معطية. هذه المتغيرة قد تكون من النوع الصحيح أو الحقيقي أو سلسلة أحرف أو من النوع حرف.
- *Donnée* : تمثل المعطية التي نريد توجيهها أو تخزينها في المتغيرة *Variable*. هذه المعطية قد تكون من النوع الصحيح أو الحقيقي أو سلسلة أحرف أو من النوع حرف.
- **:=** : تسمح بتوجيه و تخزين المعطية *Donnée* في أو داخل المتغيرة *Variable*.
- **;** : فاصلة منقوطة Point Virgule.

سوف يتم التطرق من خلال هذه الفقرة إلى كيفية إدخال عن طريق البرنامج كل من:

1. إدخال المعطيات العددية من النوع الصحيح
2. إدخال المعطيات العددية من النوع الحقيقي
3. إدخال المعطيات من النوع سلسلة حروف

4. إدخال المعطيات من النوع حرف

01.02.04 – إدخال المعطيات العددية الصحيحة:

يتم إدخال المعطيات العددية الصحيحة تبعا لطريقة إدخال المعطيات عن طريق البرنامج وفقا لطريقة الكتابة التالية:

Variable De Type Entier **:=** *Donnée De Type Entier* **;**

حيث أن:

- *Variable De Type Entier* : تمثل اسم المتغيرة التي نريد أن نوجه لها أو نخزن فيها معطية. هذه المتغيرة النوع الصحيح.
- *Donnée De Type Entier* : تمثل المعطية التي نريد توجيهها أو تخزينها في المتغيرة *Variable De Type Entier*. هذه المعطية من النوع الصحيح.
- **:=** : تسمح بتوجيه و تخزين المعطية *Donnée De Type Entier* في أو داخل المتغيرة *Variable De Type Entier*.
- **;** : فاصلة منقوطة Point Virgule.

بغرض التوضيح كيفية إدخال المعطيات من النوع الصحيح عن طريق البرنامج نأخذ الأمثلة التالية:

مثـال 01.04:

لتكن لدينا المعطية العددية الصحيحة التالية:

45000

يتم إدخال المعطية العددية من النوع الصحيح 45000 كما يلي:

a := 45000 ;

بغرض كتابة البرنامج الذي يسمح بإدخال المعطية الصحيحة 45000 عن طريق البرنامج نتبع الخطوات التالية:

1. نقوم بالتصريح بمتغيرة واحدة فقط من النوع الصحيح بغرض تخزين المعطية العددية الصحيحة 45000 و لتكن هذه المتغيرة تحمل الاسم *a*. تتم عملية التصريح بالمتغيرة الصحيحة *a* كما يلي:

Var a : integer ;

2. توجيه و تخزين المعطية الصحيحة 45000 في المتغيرة الصحيحة *a* بواسطة **:=** كما يلي:

a := 45000 ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

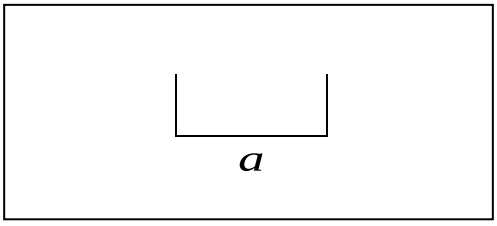
program INTRODUCTION_DES_DONNEES ;

```
var  a : integer ;
begin
    a:= 45000 ;
end.
```

عند تنفيذ البرنامج أعلاه، فإن الحاسوب يقوم بما يلي:

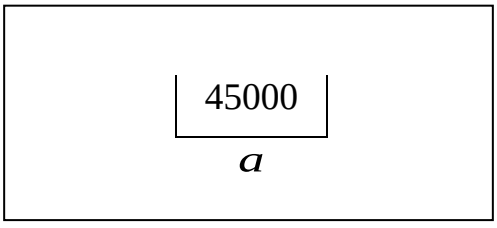
1. عند تنفيذ السطر : *var a : integer ;* فإنه يقوم تخصيص خانة في الذاكرة الحية يسميها *a* كما هو موضح أدناه:

الذاكرة الحيـة



2. عند تنفيذ السطر : *; ; a:= 45000* فإنه يقوم بتوجيه و تخزين المعطية الصحيحة 45000 في الخانة التي تم تخصيصها و حجزها في الذاكرة و التي تحمل الاسم *a* كما هو موضح أدناه:

الذاكرة الحيـة



مثـال 02.04:

لتكن لدينا المعطيات العددية الصحيحة التالية:

123 , 67 , 890 , 150 , 34 , 899

يتم إدخال المعطيات الصحيحة أعلاه كما يلي:

```
a1 := 123 ;
a2 := 67 ;
a3 := 890 ;
a4 := 150 ;
a5 := 34 ;
a6 := 899 ;
```

بغرض كتابة البرنامج الذي يسمح بإدخال المعطيات الصحيحة أعلاه عن طريق البرنامج نتبع الخطوات التالية:

1. نقوم بالتصريح بـ 6 متغيرات من النوع الصحيح بغرض تخزين الست معطيات صحيحة , 67 , 123 899 , 34 , 150 , 890 و لتكن هذه المتغيرات *a1 , a2 , a3 , a4 , a5 , a6*. تتم عملية التصريح بهذه الست متغيرات صحيحة كما يلي:

Var a1 , a2 , a3 , a4 , a5 , a6 : integer ;

2. توجيه و تخزين المعطيات الصحيحة 899 , 34 , 150 , 890 , 67 , 123 في المتغيرات الصحيحة *a1 , a2 , a3 , a4 , a5 , a6* بواسطة **:=** كما يلي:

a1 := 123 ;
a2 := 67 ;
a3 := 890 ;
a4 := 150 ;
a5 := 34 ;
a6 := 899 ;

كما يمكن كتابتها على الصيغة التالية:

a1 := 123 ; a2 := 67 ; a3 := 890 ; a4 := 150 ; a5 := 34 ; a6 := 899 ;

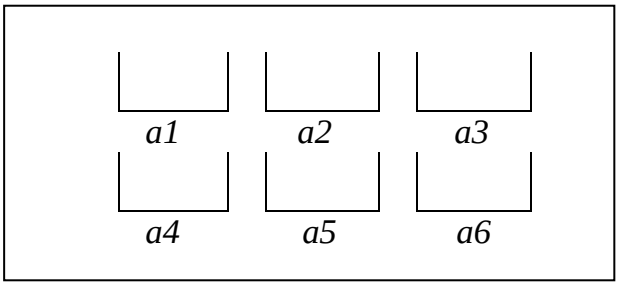
بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  a1 , a2 , a3 , a4 , a5 , a6 : integer ;
begin
    a1 := 123 ;
    a2 := 67 ;
    a3 := 890 ;
    a4 := 150 ;
    a5 := 34 ;
    a6 := 899 ;
end.
```

عند تنفيذ البرنامج أعلاه، فإن الحاسوب يقوم بما يلي:

1. عند تنفيذ السطر : *Var a1 , a2 , a3 , a4 , a5 , a6 : integer ;* فإنه يقوم تخصيص 6 خانات في الذاكرة الحية يسميها *a1 , a2 , a3 , a4 , a5 , a6* كما هو موضح أدناه:

الذاكرة الحيـة

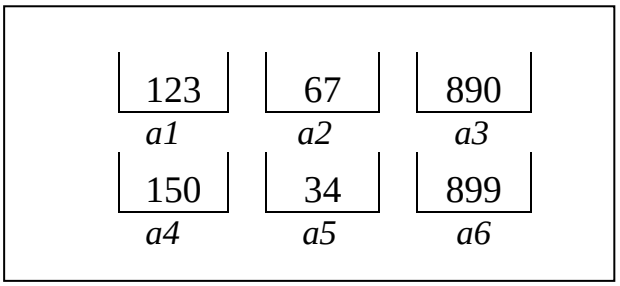


2. عند تنفيذ الأسطر التالية:

```
a1 := 123 ;
a2 := 67 ;
a3 := 890 ;
a4 := 150 ;
a5 := 34 ;
a6 := 899 ;
```

فإنه يقوم بتوجيه و تخزين المعطيات الصحيحة 899 , 34 , 150 , 890 , 67 , 123 في الخانات التي تم تخصيصها و حجزها في الذاكرة و التي تحمل الاسم *a6* , *a5* , *a4* , *a3* , *a2* , *a1* كما هو موضح أدناه:

الذاكرة الحيـة



02.02.04 – إدخـال المعطيـات العدديـة الحقيقيـة:

يتم إدخال المعطيات العددية الحقيقية تبعا لطريقة إدخال المعطيات عن طريق البرنامج وفقا لطريقة الكتابة التالية:

Variable De Type Réelle **:=** *Donnée De Type Réelle* **;**

حيث أن:

- *Variable De Type Réelle* : تمثل اسم المتغيرة التي نريد أن نوجه لها أو نخزن فيها معطية. هذه المتغيرة من النوع الحقيقي.

- *Donnée De Type Réelle* : تمثل المعطية التي نريد توجيهها أو تخزينها في المتغيرة *Variable De Type Réelle*. هذه المعطية من النوع الحقيقي.
- **:=** : تسمح بتوجيه و تخزين المعطية *Donnée De Type Réelle* في أو داخل المتغيرة *Variable De Type Réelle*.
- **;** : فاصلة منقوطة Point Virgule.

بغرض التوضيح كيفية إدخال المعطيات من النوع الحقيقي عن طريق البرنامج نأخذ الأمثلة التالية:

مثـال 03.04:

لتكن لدينا المعطية العددية الحقيقية التالية:

1250.987

يتم إدخال المعطية العددية من النوع الحقيقي 1250.987 كما يلي:

b := 1250.987 ;

بغرض كتابة البرنامج الذي يسمح بإدخال المعطية الحقيقية 1250.987 عن طريق البرنامج نتبع الخطوات التالية:

1. نقوم بالتصريح بمتغيرة واحدة فقط من النوع الحقيقي بغرض تخزين المعطية العددية الحقيقية 1250.987 و لتكن هذه المتغيرة تحمل الاسم *b*. تتم عملية التصريح بالمتغيرة الحقيقية *b* كما يلي:

Var b : real ;

2. توجيه و تخزين المعطية الحقيقية 1250.987 في المتغيرة الحقيقية *b* بواسطة **:=** كما يلي:

b := 1250.987 ;

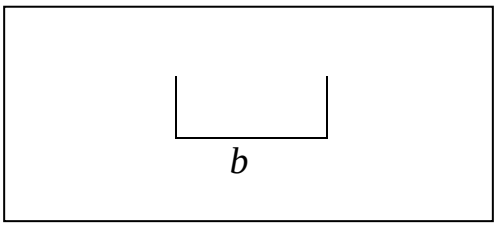
بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
var  b : real ;
begin
    b:= 1250.987 ;
end.
```

عند تنفيذ البرنامج أعلاه، فإن الحاسوب يقوم بما يلي:

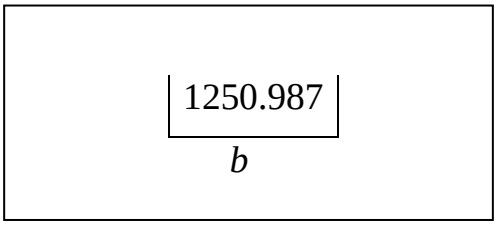
1. عند تنفيذ السطر : *var b : real ;* فإنه يقوم تخصيص خانة في الذاكرة الحية يسميها *b* كما هو موضح أدناه:

الذاكرة الحيـة



2. عند تنفيذ السطر : *b:= 1250.987 ;* فإنه يقوم بتوجيه و تخزين المعطية الحقيقية 1250.987 في الخانة التي تم تخصيصها و حجزها في الذاكرة و التي تحمل الاسم *b* كما هو موضح أدناه:

الذاكرة الحيـة



مثـال 04.04:

لتكن لدينا المعطيات العددية الحقيقية التالية:

12.563 , 0.09 , 87.55 , 670.1 , 3.4 , 5.88

يتم إدخال المعطيات الحقيقية أعلاه كما يلي:

x1 := 12.563 ;
x2 := 0.09 ;
x3 := 87.55 ;
x4 := 670.1 ;
x5 := 3.4 ;
x6 := 5.88 ;

بغرض كتابة البرنامج الذي يسمح بإدخال المعطيات الحقيقية أعلاه عن طريق البرنامج نتبع الخطوات التالية:

3. نقوم بالتصريح بـ 6 متغيرات من النوع الحقيقية بغرض تخزين الست معطيات الحقيقية , 12.563 , 5.88 , 3.4 , 670.1 , 87.55 , 0.09 و لتكن هذه المتغيرات *x6 , x5 , x4 , x3 , x2 , x1*. تتم عملية التصريح بهذه الست متغيرات الحقيقية كما يلي:

Var x1 , x2 , x3 , x4 , x5 , x6 : real ;

4. توجيه و تخزين المعطيات الحقيقية 5.88 , 3.4 , 670.1 , 87.55 , 0.09 , 12.563 في المتغيرات الحقيقية *x6 , x5 , x4 , x3 , x2 , x1* بواسطة **:=** كما يلي:

```
x1 := 12.563 ;
x2 := 0.09 ;
x3 := 87.55 ;
x4 := 670.1 ;
x5 := 3.4 ;
x6 := 5.88 ;
```

كما يمكن كتابتها على الصيغة التالية:

```
x1 := 12.563 ; x2 := 0.09 ; x3 := 87.55 ; x4 := 670.1 ; x5 := 3.4 ; x6 := 5.88 ;
```

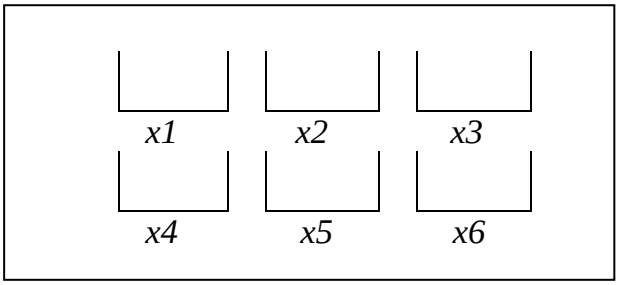
بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  x1 , x2 , x3 , x4 , x5 , x6 : real ;
begin
   x1 := 12.563 ;
   x2 := 0.09 ;
   x3 := 87.55 ;
   x4 := 670.1 ;
   x5 := 3.4 ;
   x6 := 5.88 ;
end.
```

عند تنفيذ البرنامج أعلاه، فإن الحاسوب يقوم بما يلي:

1. عند تنفيذ السطر : *Var x1 , x2 , x3 , x4 , x5 , x6 : real ;* فإنه يقوم تخصيص 6 خانات في الذاكرة الحية يسميها *x1 , x2 , x3 , x4 , x5 , x6* كما هو موضح أدناه:

الذاكرة الحيـة



2. عند تنفيذ الأسطر التالية:

```
x1 := 12.563 ;
```

x2 := 0.09 ;

x3 := 87.55 ;

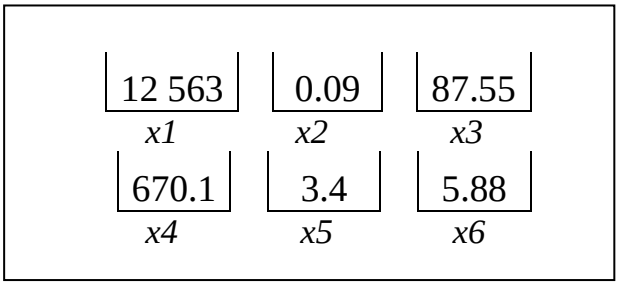
x4 := 670.1 ;

x5 := 3.4 ;

x6 := 5.88 ;

فإنه يقوم بتوجيه و تخزين المعطيات الحقيقية 5.88 , 3.4 , 670.1 , 87.55 , 0.09 , 12.563 في الخانات التي تم تخصيصها و حجزها في الذاكرة و التي تحمل الاسم *x1 , x2 , x3 , x4 , x5 , x6* , كما هو موضح أدناه:

الذاكرة الحية



03.02.04 – إدخال المعطيات سلسلة أحرف:

يتم إدخال المعطيات من النوع سلسلة أحرف تبعا لطريقة إدخال المعطيات عن طريق البرنامج وفقا لطريقة الكتابة التالية:

Variable Chaine De Caractères := ‘ Donnée Chaine De Caractères ’ ;

حيث أن:

- *Variable Chaine De Caractères* : تمثل اسم المتغيرة التي نريد أن نوجه لها أو نخزن فيها معطية. هذه المتغيرة من النوع سلسلة أحرف.
- *Donnée Chaine De Caractères* : تمثل المعطية التي نريد توجيهها أو تخزينها في المتغيرة *Variable Chaine De Caractères*. هذه المعطية من النوع سلسلة أحرف.
- **:=** : تسمح بتوجيه و تخزين المعطية *Donnée Chaine De Caractères* في أو داخل المتغيرة *Variable Chaine De Caractères*.
- ‘ ‘ : Deux quottes .
- **;** : فاصلة منقوطة Point Virgule.

بغرض التوضيح كيفية إدخال المعطيات من النوع سلسلة أحرف عن طريق البرنامج نأخذ الأمثلة التالية:

مثــال 05.04:

لتكن لدينا المعطية من النوع سلسلة أحرف التالية:

La Ville De TIARET

يتم إدخال المعطية سلسلة أحرف La Ville De TIARET كما يلي:

c := 'La Ville De TIARET' ;

بغرض كتابة البرنامج الذي يسمح بإدخال المعطية سلسلة أحرف La Ville De TIARET عن طريق البرنامج نتبع الخطوات التالية:

3. نقوم بالتصريح بمتغيرة واحدة فقط من النوع سلسلة أحرف بغرض تخزين المعطية سلسلة أحرف La Ville De TIARET و لتكن هذه المتغيرة تحمل الاسم *c*. تتم عملية التصريح بالمتغيرة الحقيقية *c* كما يلي:

Var c : string ;

4. توجيه و تخزين المعطية سلسلة أحرف La Ville De TIARET في المتغيرة *c* بواسطة **:=** كما يلي:

c := 'La Ville De TIARET' ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  c : string ;
begin
    c := 'La Ville De TIARET' ;
end.
```

مثــال 06.04:

لتكن لدينا المعطيات من النوع سلسلة أحرف التالية:

Module Informatique , note , è_çà87.55 , !§45bnk1 , nationale , 5.88

يتم إدخال المعطيات الحقيقية أعلاه كما يلي:

```
ch1 := 'Module Informatique' ;
ch 2 := 'note' ;
ch 3 := 'è_çà87.55' ;
ch 4 := '!§45bnk1' ;
ch5 := 'nationale' ;
ch 6 := '5.88' ;
```

بغرض كتابة البرنامج الذي يسمح بإدخال المعطيات سلسلة أحرف أعلاه عن طريق البرنامج نتبع الخطوات التالية:

1. نقوم بالتصريح بـ 6 متغيرات من النوع سلسلة أحرف بغرض تخزين الست معطيات سلسلة أحرف و لتكن هذه المتغيرات *ch1 , ch2 , ch3 , ch4 , ch5 , ch6*. تتم عملية التصريح بهذه الست متغيرات الحقيقية كما يلي:

Var ch1 , ch2 , ch3 , ch4 , ch5 , ch6 : string ;

2. توجيه و تخزين المعطيات سلسلة أحرف في المتغيرات سلسلة أحرف بواسطة **:=** كما يلي:

ch1 := 'Module Informatique' ;
ch 2 := 'note' ;
ch 3 := 'è_çà87.55' ;
ch 4 := '!§45bnk1' ;
ch5 := 'nationale' ;
ch 6 := '5.88' ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var ch1 , ch2 , ch3 , ch4 , ch5 , ch6 : string ;
begin
    ch1 := 'Module Informatique' ;
    ch 2 := 'note' ;
    ch 3 := 'è_çà87.55' ;
    ch 4 := '!§45bnk1' ;
    ch5 := 'nationale' ;
    ch 6 := '5.88' ;
end.
```

04.02.04 – إدخال المعطيات من النوع حرف:

يتم إدخال المعطيات من النوع حرف تبعا لطريقة إدخال المعطيات عن طريق البرنامج وفقا لطريقة الكتابة التالية:

Variable De Type Caractère := ' Donnée De Type Caractère ' ;

حيث أن:

- *Variable De Type Caractère* : تمثل اسم المتغيرة التي نريد أن نوجه لها أو نخزن فيها معطية. هذه المتغيرة من النوع سلسلة أحرف.
- *Donnée De Type Caractère* : تمثل المعطية التي نريد توجيهها أو تخزينها في المتغيرة *Variable De Type Caractère*. هذه المعطية من النوع سلسلة أحرف.
- **:=** : تسمح بتوجيه و تخزين المعطية *Donnée De Type Caractère* في أو داخل المتغيرة *Variable De Type Caractère*.
- ‘ ‘ : Deux quottes .
- **;** : فاصلة منقوطة Point Virgule.

بغرض التوضيح كيفية إدخال المعطيات من النوع حرف عن طريق البرنامج نأخذ الأمثلة التالية:

مثـال 07.04:

لتكن لدينا المعطية من النوع حرف التالية:

V

يتم إدخال المعطية سلسلة أحرف V كما يلي:

v := ‘V’ ;

بغرض كتابة البرنامج الذي يسمح بإدخال المعطية من النوع حرف V عن طريق البرنامج نتبع الخطوات التالية:

1. نقوم بالتصريح بمتغيرة واحدة فقط من النوع حرف بغرض تخزين المعطية من النوع حرف V و لتكن هذه المتغيرة تحمل الاسم *v*. تتم عملية التصريح بالمتغيرة الحقيقية *v* كما يلي:

Var v : char ;

2. توجيه و تخزين المعطية V في المتغيرة *v* بواسطة **:=** كما يلي:

v := ‘V’ ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  v : char ;
begin
    v := ‘V’ ;
end.
```

مثـال 08.04:

لتكن لدينا المعطيات من النوع حرف التالية:

M , n , ç , § , – , 5

يتم إدخال المعطيات الحقيقية أعلاه كما يلي:

```
Ch11 := 'M' ;
ch 21 := 'n' ;
ch 13 := 'ç' ;
ch 41 := '§ ;
ch15 := '–' ;
ch 61 := '5' ;
```

بغرض كتابة البرنامج الذي يسمح بإدخال المعطيات من النوع حرف أعلاه عن طريق البرنامج نتبع الخطوات التالية:

1. نقوم بالتصريح بـ 6 متغيرات من النوع حرف بغرض تخزين الست معطيات من النوع حرف و لتكن هذه المتغيرات *ch11 , ch21 , ch13 , ch41 , ch15 , ch61*. تتم عملية التصريح بهذه الست متغيرات كما يلي:

```
Var  ch11 , ch21 , ch13 , ch41 , ch15 , ch61 : char ;
```

2. توجيه و تخزين المعطيات في المتغيرات بواسطة **:=** كما يلي:

```
Ch11 := 'M' ;
ch 21 := 'n' ;
ch 13 := 'ç' ;
ch 41 := '§ ;
ch15 := '–' ;
ch 61 := '5' ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  ch11 , ch21 , ch13 , ch41 , ch15 , ch61 : char ;
begin
    Ch11 := 'M' ;
    ch 21 := 'n' ;
    ch 13 := 'ç' ;
    ch 41 := '§ ;
    ch15 := '–' ;
    ch 61 := '5' ;
end.
```

03.04 – إدخـال المعطيـات عن طريـق لوحـة المفاتيـح:

Introduction des données par le clavier

إضافة إلى طريقة إدخال المعطيات بأنواعها عن طريق البرنامج و التي مضمونها أن المعطيات تكون موجودة في البرنامج أي يتم توجهيها و تخزينها في المتغيرات الموافقة لها عبر البرنامج، هناك طريقة أخرى لإدخال المعطيات تتمثل في إدخال المعطيات عن طريق لوحة المفاتيح (Par Le Clavier). مضمون هذه الطريقة هو أنه يتم إدخال المعطيات بعد تنفيذ البرنامج حيث يطلب الحاسوب من المستخدم إدخال المعطيات ليقوم هذا الأخير بإدخالها عن طريق لوحة المفاتيح. يتم إدخال المعطيات تبعا لهذه الطريقة وفقا لطريقة الكتابة التالية Suivant La Syntaxe Suivante :

Read (*Variable* ***) ;***

أو

Readln (*Variable* ***) ;***

حيث أن:

- *Variable* : تمثل اسم المتغيرة التي نريد أن نوجه لها أو نخزن فيها معطية. هذه المتغيرة قد تكون من النوع الصحيح أو الحقيقي أو سلسلة أحرف أو من النوع حرف.
- *Read* : تمثل Identificateur Prédéfinis تسمح بقراءة المعطية التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable*.
- **;** : فاصلة منقوطة Point Virgule.

سوف يتم التطرق من خلال هذه الفقرة إلى كيفية إدخال عن طريق لوحة المفاتيح كل من:

1. المعطيات العددية من النوع الصحيح
2. المعطيات العددية من النوع الحقيقي
3. المعطيات من النوع سلسلة حروف
4. المعطيات من النوع حرف

01.03.04 – إدخـال المعطيـات الصحيحـة:

يتم إدخال المعطيات من النوع الصحيح تبعا لطريقة إدخال المعطيات عن طريق لوحة المفاتيح وفقا لطريقة الكتابة التالية:

Read (*Variable De Type Entier* ***) ;***

أو

Readln (*Variable De Type Entier* ***) ;***

حيث أن:

- *Variable De Type Entier* : تمثل اسم المتغيرة من النوع الصحيح التي نريد أن نوجه لها أو نخزن فيها معطية من النوع الصحيح.
- *Read* : تمثل Identificateur Prédéfinis تسمح بقراءة المعطية من النوع الصحيح التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable De Type Entier*.
- *Readln* : تمثل Identificateur Prédéfinis au Pascal تسمح بقراءة المعطية من النوع الصحيح التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable De Type Entier*.

ملاحظـة 01.04 :

كل من *Read* و *Readln* تسمح بقراءة المعطيات بجميع أنواعها إلا أن هناك اختلاف طفيف بينهما سوف يتم التطرق إليه لاحقا.

بغرض توضيح كيفية إدخال المعطيات عن طريق لوحة المفاتيح باستخدام Readln نأخذ المثال التالي:

مثـال 09.04:

لتكن لدينا المعطية العددية من النوع الصحيح التالية:

98000

بغرض كتابة البرنامج الذي يسمح بإدخال المعطية الصحيحة 98000 عن طريق لوحة المفاتيح نتبع الخطوات التالية:

1. نقوم بالتصريح بمتغيرة واحدة فقط من النوع الصحيح بغرض تخزين المعطية العددية الصحيحة 98000 و لتكن هذه المتغيرة تحمل الاسم *a*. تتم عملية التصريح بالمتغيرة الصحيحة *a* كما يلي:

Var a : integer ;

2. إدخال المعطية الصحيحة 98000 عن طريق لوحة المفاتيح كما يلي:

Readln (a) ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

program INTRODUCTION_DES_DONNEES ;

```
var  a : integer ;
begin
    Readln ( a ) ;
end.
```

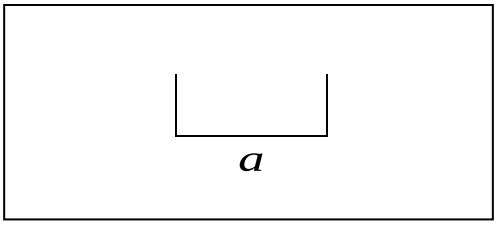
عند تنفيذ البرنامج أعلاه، فإن الحاسوب يقوم بما يلي:

1. عند تنفيذ السطر :

var a : integer ;

فإن الحاسوب يقوم بتخصيص خانة في الذاكرة الحية يسميها *a* كما هو موضح أدناه:

الذاكرة الحيـة



2. عند تنفيذ السطر :

Readln (a) ;

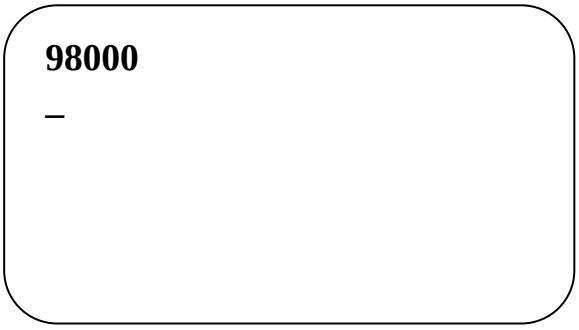
فإننا نحصل على شاشة فارغة تحتوي على _Le Cursseur في العمود الأول للسطر الأول كما هو موضح على الشاشة أدناه:

الشاشة

يقوم الحاسوب بانتظار المستخدم بإدخال المعطية الصحيحة 98000 عن طريق لوحة المفاتيح.

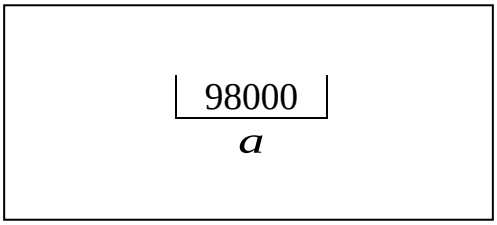
3. يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة القيمة 98000 ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



4. بعد إدخال المعطية 98000 عن طريق لوحة المفاتيح يقوم الحاسوب بتخزينها في الخانة a التي تم حجزها على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحيـة



مثـال 10.04:

لتكن لدينا المعطيات العددية الصحيحة التالية:

1250 , 4567 , 12000 , 6789 , 23550

بغرض كتابة البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح نتبع الخطوات التالية:

1. نقوم بالتصريح بـ 5 متغيرات من النوع الصحيح و لتكن *v1 , v2 , v3 , v4 , v5* بغرض تخزين المعطيات أعلاه. عملية التصريح تتم كما يلي:

```
Var  v1 , v2 , v3 , v4 , v5 : integer ;
```

2. إدخال المعطيات أعلاه عن طريق لوحة المفاتيح يتم ذلك باستخدام read أو readln كما يلي:

```
readln ( v1 ) ;
readln ( v2 ) ;
readln ( v3 ) ;
readln ( v4 ) ;
readln ( v5 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  v1 , v2 , v3 , v4 , v5 : integer ;
```

```
begin
    readln ( v1 ) ;
    readln ( v2 ) ;
    readln ( v3 ) ;
    readln ( v4 ) ;
    readln ( v5 ) ;

end.
```

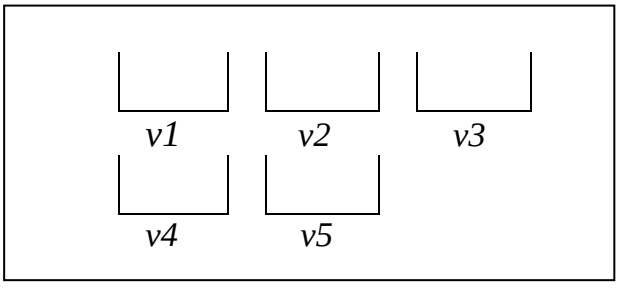
عند تنفيذ البرنامج أعلاه، فإن الحاسوب يقوم بما يلي:

1. عند تنفيذ السطر :

Var v1 , v2 , v3 , v4 , v5 : integer ;

فإن الحاسوب يقوم بتخصيص 6 خانات في الذاكرة الحية يسميها *v1 , v2 , v3 , v4 , v5* كما هو موضح أدناه:

الذاكرة الحيـة



2. عند تنفيذ السطر :

readln (v1) ;

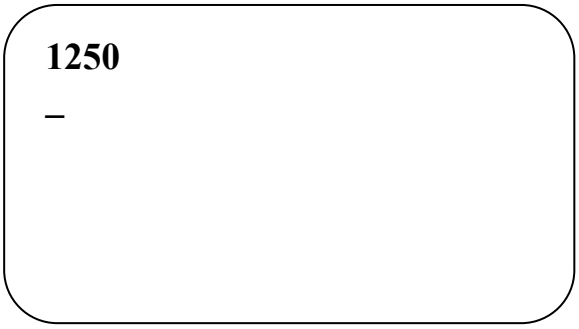
فإننا نحصل على شاشة فارغة تحتوي على _Le Cursseur في العمود الأول للسطر الأول كما هو موضح على الشاشة أدناه:

الشاشة

يقوم الحاسوب بانتظار المستخدم بإدخال المعطية الأولى عن طريق لوحة المفاتيح و التي سوف يخزنها في الخانة التي تحمل الاسم *v1*.

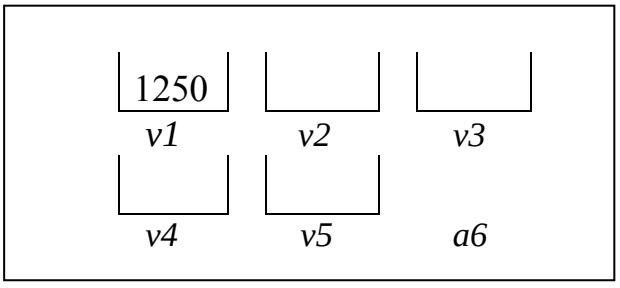
يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة المعطية الأولى 1250 ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



بعد إدخال المعطية 1250 عن طريق لوحة المفاتيح يقوم الحاسوب بتخزينها في الخانة *v1* التي تم حجزها على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحية



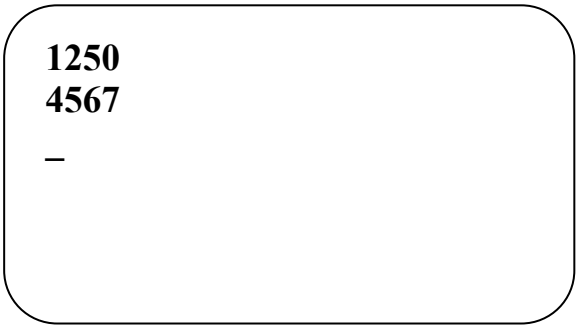
3. عند تنفيذ السطر :

readln (v2) ;

يقوم الحاسوب بانتظار المستخدم بإدخال المعطية الثانية عن طريق لوحة المفاتيح و التي سوف يخزنها في الخانة التي تحمل الاسم *v2*.

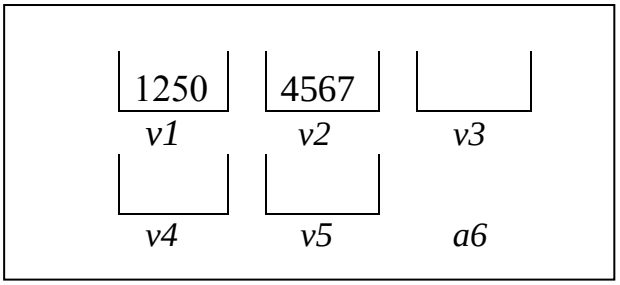
يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة المعطية الثانية 4567 ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



بعد إدخال المعطية 4567 عن طريق لوحة المفاتيح يقوم الحاسوب بتخزينها في الخانة *v2* التي تم حجزها على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحيـة



4. و هكذا إلى غاية تنفيذ السطر الأخير

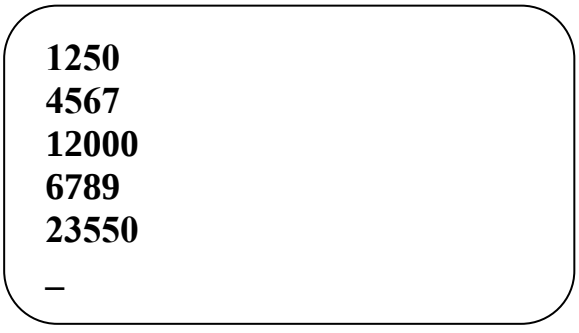
5. عند تنفيذ السطر :

readln (v5) ;

يقوم الحاسوب بانتظار المستخدم بإدخال المعطية الخامسة 23550 عن طريق لوحة المفاتيح و التي سوف يخزنها في الخانة التي تحمل الاسم *v5*.

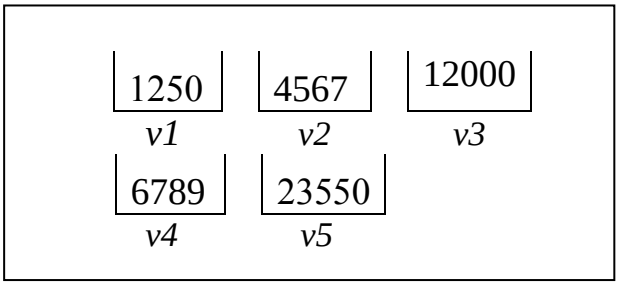
يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة المعطية الخامسة 23550 ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



بعد إدخال المعطية 23550 عن طريق لوحة المفاتيح يقوم الحاسوب بتخزينها في الخانة *v5* التي تم حجزها على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحيـة



02.03.04 – إدخـال المعطيـات الحقيقيـة:

يتم إدخال المعطيات من النوع الحقيقي تبعا لطريقة إدخال المعطيات عن طريق لوحة المفاتيح وفقا لطريقة الكتابة التالية:

Read (*Variable De Type Réelle* ***) ;***

أو

Readln (*Variable De Type Réelle* ***) ;***

حيث أن:

- *Variable De Type Réelle* : تمثل اسم المتغيرة من النوع الحقيقي التي نريد أن نوجه لها أو نخزن فيها معطية من النوع الحقيقي.
- *Read* : تمثل Identificateur Prédéfinis تسمح بقراءة المعطية من النوع الحقيقي التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable De Type Réelle*.
- *Readln* : تمثل Identificateur Prédéfinis تسمح بقراءة المعطية من النوع الحقيقي التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable De Type Réelle*.

بغرض توضيح كيفية إدخال المعطيات عن طريق لوحة المفاتيح باستخدام Readln نأخذ المثال التالي:

مثـال 11.04:

لتكن لدينا المعطية العددية من النوع الحقيقي التالية:

1250.345

بغرض كتابة البرنامج الذي يسمح بإدخال المعطية الحقيقية أعلاه عن طريق لوحة المفاتيح نتبع الخطوات التالية:

1. نقوم بالتصريح بمتغيرة واحدة فقط من النوع الحقيقي بغرض تخزين المعطية العددية الحقيقية 1250.345 و لتكن هذه المتغيرة تحمل الاسم *x*. تتم عملية التصريح بالمتغيرة الحقيقية *x* كما يلي:

Var x : real ;

2. إدخال المعطية الصحيحة 1250.345 عن طريق لوحة المفاتيح كما يلي:

Readln (x) ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
var  x : real ;
begin
    Readln ( x ) ;
end.
```

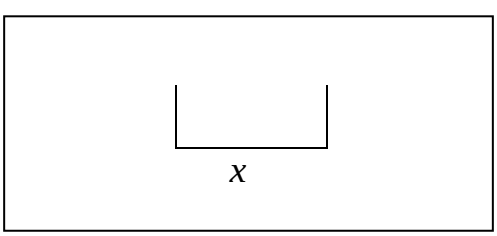
عند تنفيذ البرنامج أعلاه، فإن الحاسوب يقوم بما يلي:

1. عند تنفيذ السطر :

var x : real ;

فإن الحاسوب يقوم بتخصيص خانة في الذاكرة الحية يسميها *x* كما هو موضح أدناه:

الذاكرة الحيـة

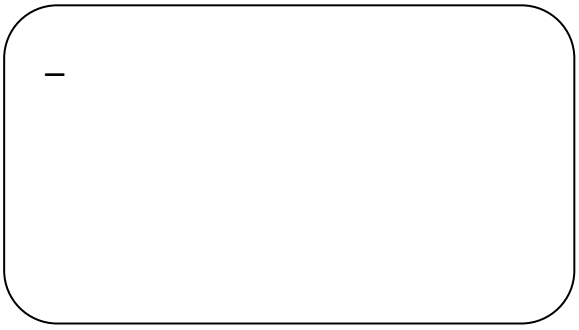


2. عند تنفيذ السطر :

Readln (x) ;

فإننا نحصل على شاشة فارغة تحتوي على _Le Cursseur في العمود الأول للسطر الأول كما هو موضح على الشاشة أدناه:

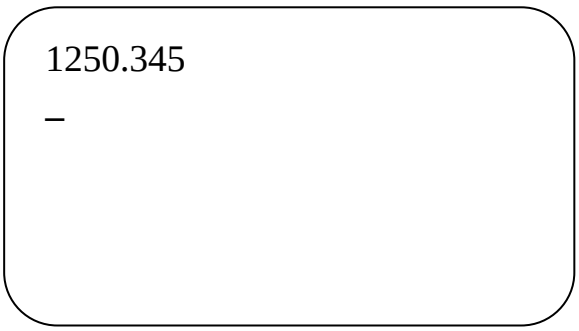
الشاشة



يقوم الحاسوب بانتظار المستخدم بإدخال المعطية الحقيقية 1250.345 عن طريق لوحة المفاتيح.

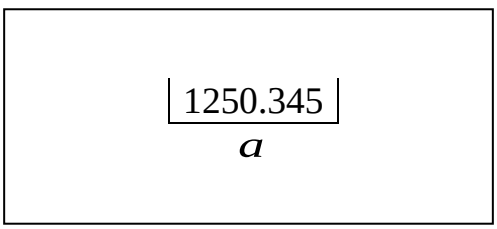
3. يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة القيمة 1250.345 ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



4. بعد إدخال المعطية 1250.345 عن طريق لوحة المفاتيح يقوم الحاسوب بتخزينها في الخانة x التي تم حجزها على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحيـة



مثـال 12.04:

لتكن لدينا المعطيات العددية الحقيقية التالية:

0.135 , 45.55 , 120.99

بغرض كتابة البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح نتبع الخطوات التالية:

1. نقوم بالتصريح بـ 3 متغيرات من النوع الحقيقي و لتكن *r1 , r2 , r3* بغرض تخزين المعطيات أعلاه.
عملية التصريح تتم كما يلي:

```
Var  r1 , r2 , r3  : real ;
```

2. إدخال المعطيات أعلاه عن طريق لوحة المفاتيح يتم ذلك باستخدام read أو readln كما يلي:

```
readln ( r1 ) ;
readln ( r2 ) ;
readln ( r3 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  r1 , r2 , r3  : real ;
begin
    readln ( r1 ) ;
    readln ( r2 ) ;
    readln ( r3 ) ;

end.
```

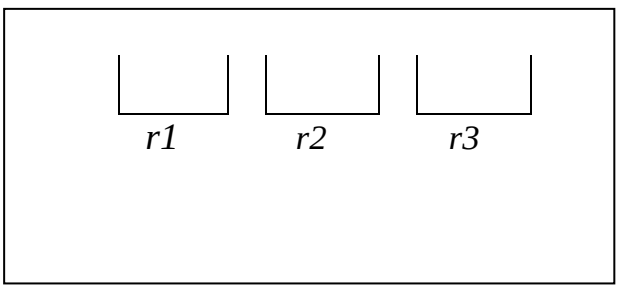
عند تنفيذ البرنامج أعلاه، فإن الحاسوب يقوم بما يلي:

1. عند تنفيذ السطر :

```
Var  r1 , r2 , r3  : real ;
```

فإن الحاسوب يقوم بتخصيص 3 خانات في الذاكرة الحية يسميها *r1 , r2 , r3* كما هو موضح أدناه:

الذاكرة الحيـة

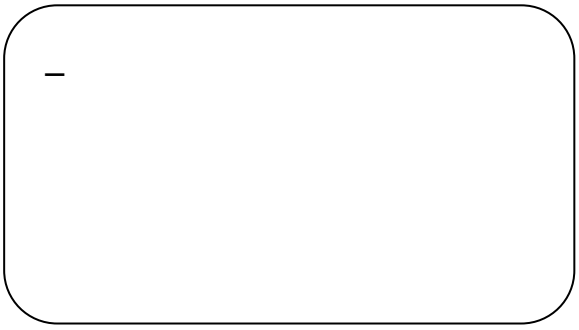


2. عند تنفيذ السطر :

```
readln ( r1 ) ;
```

فإننا نحصل على شاشة فارغة تحتوي على _Le Cursseur في العمود الأول للسطر الأول كما هو موضح على الشاشة أدناه:

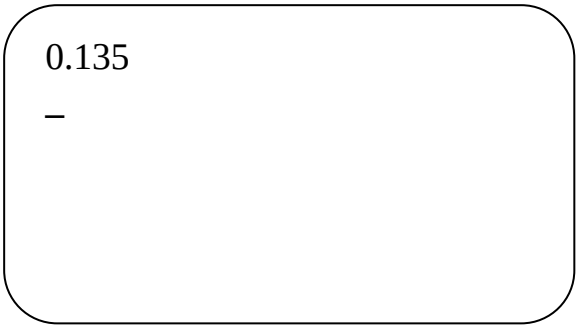
الشاشة



يقوم الحاسوب بانتظار المستخدم بإدخال المعطية الأولى عن طريق لوحة المفاتيح و التي سوف يخزنها في الخانة التي تحمل الاسم *r1*.

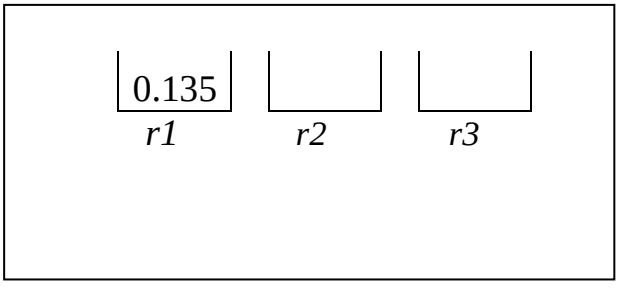
يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة المعطية الأولى 0.135 ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



بعد إدخال المعطية 0.135 عن طريق لوحة المفاتيح يقوم الحاسوب بتخزينها في الخانة *r1* التي تم حجزها على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحيـة



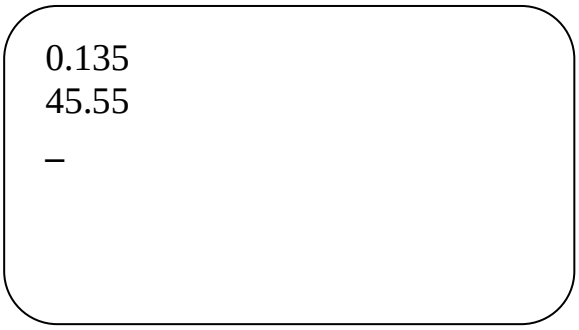
3. عند تنفيذ السطر :

readln (r2) ;

يقوم الحاسوب بانتظار المستخدم بإدخال المعطية الثانية 45.55 عن طريق لوحة المفاتيح و التي سوف يخزنها في الخانة التي تحمل الاسم *r2*.

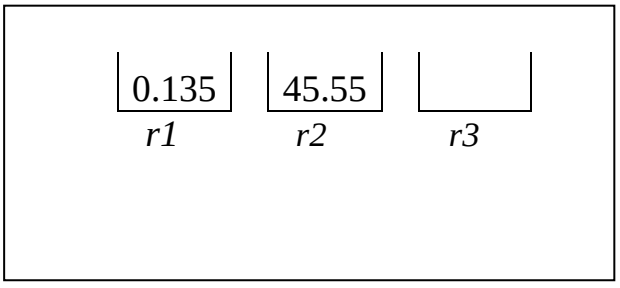
يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة المعطية الثانية 45.55 ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



بعد إدخال المعطية 45.55 عن طريق لوحة المفاتيح يقوم الحاسوب بتخزينها في الخانة *r2* التي تم حجزها على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحيـة



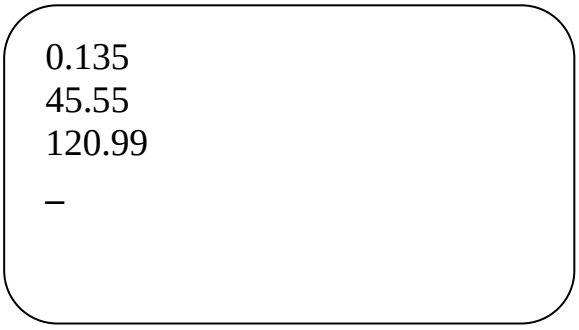
4. عند تنفيذ السطر :

readln (r3) ;

يقوم الحاسوب بانتظار المستخدم بإدخال المعطية الخامسة 120.99 عن طريق لوحة المفاتيح و التي سوف يخزنها في الخانة التي تحمل الاسم *r3*.

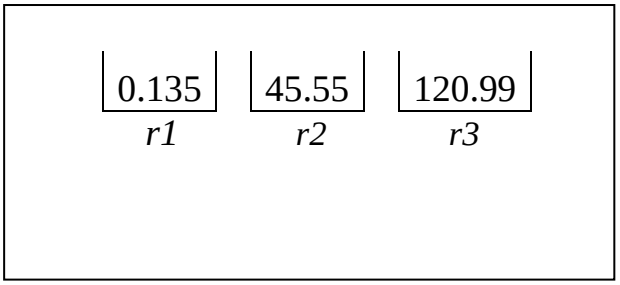
يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة المعطية الخامسة 120.99 ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



بعد إدخال المعطية 120.99 عن طريق لوحة المفاتيح يقوم الحاسوب بتخزينها في الخانة *v5* التي تم حجزها على مستوى الذاكرة كما هو موضح أدناه:

الذاكرة الحيـة



ملاحظـة 02.04 :

يمكن توجيه معطية عددية من النوع الصحيح إلى متغيرة عددية من النوع الحقيقي غير أنه من غير الممكن توجيه معطية عددية من النوع الحقيقي إلى متغيرة من النوع الصحيح.

03.03.04 – إدخـال المعطيـات سلسلـة أحـرف:

يتم إدخال المعطيات من النوع سلسلة أحرف تبعا لطريقة إدخال المعطيات عن طريق لوحة المفاتيح وفقا لطريقة الكتابة التالية:

Read (*Variable De Type Chaine De Caractères* ***) ;***

أو

Readln (*Variable De Type Chaine De Caractères* ***) ;***

حيث أن:

- *Variable De Type Chaine De Caractères* : تمثل اسم المتغيرة من النوع سلسلة أحرف التي نريد أن نوجه لها أو نخزن فيها معطية من النوع سلسلة أحرف.
- *Read* : تمثل Identificateur Prédéfinis تسمح بقراءة المعطية من النوع سلسلة أحرف التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable De Type Chaine De Caractères*.
- *Readln* : تمثل Identificateur Prédéfinis تسمح بقراءة المعطية من النوع سلسلة أحرف التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable De Type Chaine De Caractère*.

بغرض توضيح كيفية إدخال المعطيات سلسلة أحرف عن طريق لوحة المفاتيح باستخدام Readln نأخذ المثال التالي:

مثـال 13.04:

لتكن لدينا المعطية من النوع سلسلة أحرف التالية:

Module INFORMATIQUE

بغرض كتابة البرنامج الذي يسمح بإدخال المعطية سلسلة أحرف Module INFORMATIQUE عن طريق لوحة المفاتيح نتبع الخطوات التالية:

3. نقوم بالتصريح بمتغيرة واحدة فقط من النوع سلسلة أحرف بغرض تخزين المعطية أعلاه و لتكن هذه المتغيرة تحمل الاسم *mod*. تتم عملية التصريح بالمتغيرة الصحيحة *mod* كما يلي:

```
Var  mod : string ;
```

4. إدخال المعطية سلسلة أحرف أعلاه عن طريق لوحة المفاتيح كما يلي:

```
Readln ( mod ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

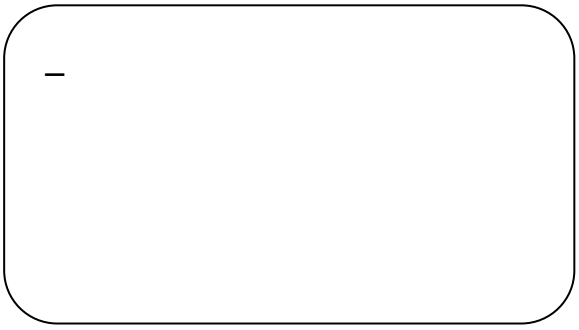
```
program INTRODUCTION_DES_DONNEES ;
Var  mod : string[20] ;
begin
    Readln ( mod ) ;
end.
```

1. عند تنفيذ السطر :

```
Readln ( mod ) ;
```

فإننا نحصل على شاشة فارغة تحتوي على _Le Cursseur في العمود الأول للسطر الأول كما هو موضح على الشاشة أدناه:

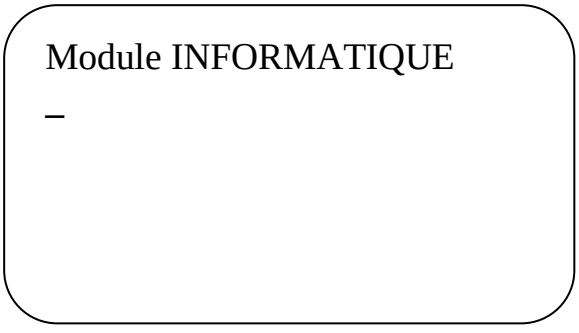
الشاشة



يقوم الحاسوب بانتظار المستخدم بإدخال المعطية Module INFORMATIQUE عن طريق لوحة المفاتيح.

يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة Module INFORMATIQUE ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



مثـال 14.04:

لتكن لدينا المعطيات من النوع سلسلة أحرف التالية:

STATISTIQUE , SIHEM , La Ville De Tiaret , Sougueur , 234(:/§*

بغرض كتابة البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح نتبع الخطوات التالية:

1. نقوم بالتصريح بـ 5 متغيرات من النوع سلسلة أحرف و لتكن *ch1 , ch2 , ch3 , ch4 , ch5* بغرض تخزين المعطيات أعلاه. عملية التصريح تتم كما يلي:

```
Var  ch1 , ch2 , ch3 , ch4 , ch5 : string[15] ;
```

2. إدخال المعطيات أعلاه عن طريق لوحة المفاتيح يتم ذلك باستخدام read أو readln كما يلي:

```
readln (ch1 ) ;
readln (ch2 ) ;
readln (ch3 ) ;
```

```
readln (ch4 ) ;
readln (ch5 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  ch1 , ch2 , ch3 , ch4 , ch5 : string[15] ;
begin
    readln (ch1 ) ;
    readln (ch2 ) ;
    readln (ch3 ) ;
    readln (ch4 ) ;
    readln (ch5 ) ;

end.
```

04.03.04 – إدخـال المعطيـات من النـوع حـرف:

يتم إدخال المعطيات من النوع حرف تبعا لطريقة إدخال المعطيات عن طريق لوحة المفاتيح وفقا لطريقة الكتابة التالية:

Read (*Variable De Type Caractère* ***) ;***

أو

Readln (*Variable De Type Caractère* ***) ;***

حيث أن:

- *Variable De Type Caractère* : تمثل اسم المتغيرة من النوع حرف التي نريد أن نوجه لها أو نخزن فيها معطية من النوع حرف.
- *Read* : تمثل Identificateur Prédéfinis تسمح بقراءة المعطية من النوع حرف التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable De Type Caractère*.
- *Readln* : تمثل Identificateur Prédéfinis تسمح بقراءة المعطية من النوع حرف التي سوف يتم إدخالها عن طريق لوحة المفاتيح لتقوم بعد ذالك بتخزينها في المتغيرة *Variable De Type Caractère*.

بغرض توضيح كيفية إدخال المعطيات من النوع حرف عن طريق لوحة المفاتيح باستخدام Readln نأخذ المثال التالي:

مثـال 15.04:

لتكن لدينا المعطية من النوع حرف التالية:

Z

بغرض كتابة البرنامج الذي يسمح بإدخال المعطية من النوع حرف Z عن طريق لوحة المفاتيح نتبع الخطوات التالية:

1. نقوم بالتصريح بمتغيرة واحدة فقط من النوع حرف بغرض تخزين المعطية أعلاه و لتكن هذه المتغيرة تحمل الاسم *z*. تتم عملية التصريح بالمتغيرة الصحيحة *z* كما يلي:

```
Var  z : char ;
```

2. إدخال المعطية سلسلة أحرف أعلاه عن طريق لوحة المفاتيح كما يلي:

```
Readln ( z ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  z : char ;
begin
    Readln ( z ) ;
end.
```

1. عند تنفيذ السطر :

```
Readln ( z ) ;
```

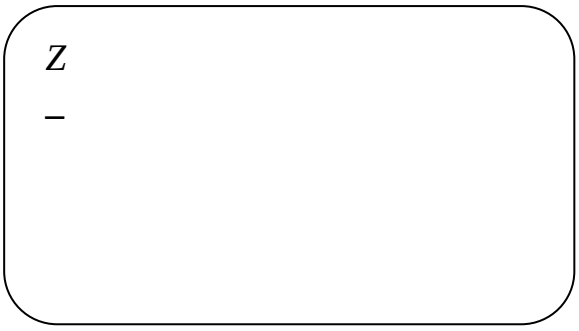
فإننا نحصل على شاشة فارغة تحتوي على _Le Cursseur في العمود الأول للسطر الأول كما هو موضح على الشاشة أدناه:

الشاشة

يقوم الحاسوب بانتظار المستخدم بإدخال المعطية Z عن طريق لوحة المفاتيح.

يقوم مستخدم الحاسوب عن طريق لوحة المفاتيح بكتابة Z ثم يضغط على اللمسة La Touche Entrer كما هو موضح في الشاشة أدناه:

الشاشة



مثـال 16.04:

لتكن لدينا المعطيات من النوع حرف التالية:

S , I , a , 5 , *

بغرض كتابة البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح نتبع الخطوات التالية:

1. نقوم بالتصريح بـ 5 متغيرات من النوع سلسلة أحرف و لتكن *ch1 , ch2 , ch3 , ch4 , ch5* بغرض تخزين المعطيات أعلاه. عملية التصريح تتم كما يلي:

```
Var  ch1 , ch2 , ch3 , ch4 , ch5 : char ;
```

2. إدخال المعطيات أعلاه عن طريق لوحة المفاتيح يتم ذلك باستخدام read أو readln كما يلي:

```
readln (ch1 ) ;
readln (ch2 ) ;
readln (ch3 ) ;
readln (ch4 ) ;
readln (ch5 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var  ch1 , ch2 , ch3 , ch4 , ch5 : char ;
begin
    readln (ch1 ) ;
    readln (ch2 ) ;
    readln (ch3 ) ;
    readln (ch4 ) ;
```

```
readln (ch5 ) ;

end.
```

04.04 – إدخال المعطيات مع التعليق: Introduction Des Données Avec Commentaire

قد نرغب في إدخال قيمة المتغيرة الصحيحة *c* بكيفية أكثر وضوحا على الشاشة كما يلي:

Donnez La Valeur De La Variable Entière c :

بغرض إدخال قيمة المتغيرة الصحيحة *c* على الشكل الموضح أعلاه نستعين بتعليمة النشر write[1] كما يلي:

```
Write ( Donnez La Valeur De La Variable Entière c : ) ;
Read (c) ;
```

البرنامج الذي يسمح بذالك هو التالي:

```
program INTRODUCTION_DES_DONNEES ;
Var c : integer ;
Begin
  Write ( Donnez La Valeur De La Variable Entière c : ) ;
  Read (c) ;
end.
```

[1] لمزيد من التفصيل حول L'Identificateur ***write*** أنظر الفصل السادس

تمـاريـــن الفصـــل

✍ التمـــريـــن 01.04:

1. قدم مختلف الطرق المستخدمة لإدخال المعطيات.

✍ التمـــريـــن 02.04:

1. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات عن طريق البرنامج.

2. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات عن لوحة المفاتيح.

✍ التمـــريـــن 03.04:

1. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات الصحيحة عن طريق البرنامج.

2. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات الصحيحة عن لوحة المفاتيح.

✍ التمـــريـــن 04.04:

لتكن المعطيات العددية الصحيحة التالية:

9000 , 11578 , 56000 , 340707

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـــريـــن 05.04:

لتكن لدينا معطيات التمرين 04.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح (Introduction Des Données Par Clavier).

✍ التمـــريـــن 06.04:

لتكن المعطيات التالية:

234 , 56789 , 78990 , 787878 , 45 , 123 , 5670

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـريـن 07.04:

لتكن لدينا معطيات التمرين 06.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح (Introduction Des Données Par Clavier).

✍ التمـريـن 08.04:

لتكن لدينا المعطية الصحيحة التالية:

$$A = 150000$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال قيمة المتغيرة A عن طريق لوحة المفاتيح. شريطة أن عملية إدخال قيمة المتغيرة A تكون كما هو موضح على الشاشة أدناه:

الشاشة

```
Donnez La Valeur De A : _
```

✍ التمـريـن 09.04:

لتكن لدينا المعطيات العددية الصحيحة التالية:

$$A = 1250$$

$$B = 2478$$

$$C = 123$$

$$CU = 4567$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـريـن 10.04:

لتكن لدينا معطيات التمرين 09.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال قيمة المتغيرات أعلاه عن طريق لوحة المفاتيح. شريطة أن عملية الإدخال تتم كما هو موضح على الشاشة أدناه:

الشاشة

```
Donnez La Valeur De La Variable A :
Donnez La Valeur De La Variable B :
Donnez La Valeur De La Variable C :
Donnez La Valeur De La Variable CU :
```

✍ التمـريـن 11.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  A , A1 , SOM , P , B : integer ;
begin
     A := 56780 ;
     B := 1238 ;
     SOM := 34567 ;
     A := 234000 ;
     P := 1200 ;
     A := 2500 ;
     som  := 12000 ;
end.
```

- **المطلـوب:**

1. حدد ماذا يقوم البرنامج أعلاه.

✍ التمـريــن 12.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  x , y , xy , v , w : integer ;
begin
      a = 56780 ;
      y := 1238
      v := '34567' ;
      w := 234000 ;
      xy = 4567 ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.

2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريــن 13.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  v1 , v5 , c , g , h : integer ;
begin
      v := 1230 ;
      v1 := 45 ;
      h = 667 ;
      g := 45678
      s  := 45000 ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريـن 14.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  v1 , v5 , c , v , g , h : integer ;
begin
      v := 12.30 ;
      v1 := 45 ;
      h := 667 ;
      g := 45.678;
      s  := 450.78 ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريـن 15.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Clavier ;
var  a , b , c , d , e , f : integer ;
begin
      readln ( a ) ;
      readln ( b ) ;
      readln ( c ) ;
      readln ( d ) ;
      readln ( e ) ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.
3. ماذا يقوم البرنامج المتحصل عليه في السؤال 2 .

✍ التمـريـن 16.04:

1. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات الحقيقية عن طريق البرنامج.
2. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات الحقيقية عن طريق لوحة المفاتيح.

✍ التمـريـن 17.04:

لتكن المعطيات العددية الحقيقية التالية:

129.798 , 0.6598 , 6578.48 , 321.255

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـريـن 18.04:

لتكن لدينا معطيات التمرين 17.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح (Introduction Des Données Par Clavier).

✍ التمـريـن 19.04:

لتكن المعطيات التالية:

98.006 , 8809.089 , 111.789 , 991.78 , 4.5 , 12.333 , 0.5670

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

التمـريــن 20.04:

لتكن لدينا معطيات التمرين 19.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح (Introduction Des Données Par Clavier).

التمـريــن 21.04:

لتكن لدينا المعطية الحقيقية التالية:

$$X = 4500.567$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال قيمة المتغيرة X عن طريق لوحة المفاتيح. شريطة أن عملية إدخال قيمة المتغيرة X تكون كما هو موضح على الشاشة أدناه:

الشاشة

```
Donnez La Valeur De La Variable X SVP : _
```

التمـريــن 22.04:

لتكن لدينا المعطيات العددية الحقيقية التالية:

$$X = 12.50$$
$$Y = 0.124$$
$$Z = 123.0001$$
$$W = 900.145$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـــريـــن 23.04:

لتكن لدينا معطيات التمرين 22.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال قيم المتغيرات أعلاه عن طريق لوحة المفاتيح. شريطة أن عملية الإدخال تتم كما هو موضح على الشاشة أدناه:

الشاشة

```
Donnez La Valeur De La Variable X :
Donnez La Valeur De La Variable Y :
Donnez La Valeur De La Variable Z :
Donnez La Valeur De La Variable W :
```

✍ التمـــريـــن 24.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  al, x , m , j , p , s : real ;
begin
    al := 1.238 ;
    m := 345.67 ;
    x := 234.000 ;
    p := 0.12 ;
    j := 25.007 ;
    s  := 12.88 ;
end.
```

- **المطلـوب:**

1. حدد ماذا يقوم البرنامج أعلاه.

✍ التمـــريـــن 25.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  x , y , xy , v , w : real ;
begin
      a = 56.780 ;
      y := 12.38
      v := '345.67' ;
      w := 2340.99 ;
      xy = 45.67 ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريـن 26.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  v1 , v5 , c , g , h : real ;
begin
      v := 1.230 ;
      v1 := 4.5 ;
      h = 66.7 ;
      g := 456.78
      s  := 450.99 ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريـن 27.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  v1 , v5 , c , v , g , h : real ;
begin
      v := 12.30 ;
      v1 := 45 ;
      h := 667 ;
      g := 45.678 ;
      s  := 450.78 ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريـن 28.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Clavier ;
var  a , b , c , d , e , f : reeal ;
begin
      readln ( a ) ;
      readln ( b ) ;
      readln ( c ) ;
      readln ( d ) ;
      readln ( e ) ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.
3. ماذا يقوم البرنامج المتحصل عليه في السؤال 2 .

✍ التمـريـن 29.04:

لتكن المعطيات التالية:

$$FAR = 0.1098$$

$$ZT = 167.876$$

$$CF = 90.4598$$

$$F = 345.123$$

$$F = 65.0976$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج.
2. أكتب البرنامج الذي يسمح بإدخال قيم المتغيرات أعلاه عن طريق لوحة المفاتيح.

✍ التمـريـن 30.04:

لتكن المعطيات التالية:

$$F = 7005.7896$$

$$F1 = 0.09012$$

$$F2 = 4509.789$$

$$F45 = 678.009$$

$$FRT = 1259.98702$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج.
2. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح.

✍ التمـريـن 31.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  e , r2 , r21 , X , L : real ;
begin
    L := 5607.80 ;
    L := 100.238 ;
    e := 0.34567 ;
    r2 := 23.4000 ;
```

```
X := 10.2008 ;
e := 2500.789 ;
r21  := 0.1002 ;
end.
```

- **المطلـوب:**

1. حدد القيمة النهائية لكل متغيرة.

✍ التمـريـن 32.04:

1. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات من النوع سلسلة أحرف عن طريق البرنامج.

2. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات من النوع سلسلة أحرف عن طريق لوحة المفاتيح.

✍ التمـريـن 33.04:

لتكن المعطيات من النوع سلسلة أحرف التالية:

Livre , cahier , statistique , tr567*%£ , 785JH , 6789

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـريـن 34.04:

لتكن لدينا معطيات التمرين 33.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح (Introduction Des Données Par Clavier).

✍ التمـريـن 35.04:

لتكن المعطيات التالية:

STATISTIQUE , SIHEM , La Ville De Tiaret , Sougueur , 234(:/§*

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـريـن 36.04:

لتكن لدينا معطيات التمرين 35.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح (Introduction Des Données Par Clavier).

✍ التمـريـن 37.04:

لتكن لدينا المعطية التالية:

T= LA VILLE DE TIARET EST BELLE

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال قيمة المتغيرة T عن طريق لوحة المفاتيح. شريطة أن عملية إدخال قيمة المتغيرة T تكون كما هو موضح على الشاشة أدناه:

الشاشة

```
Donnez La Valeur De La Variable T SVP : _
```

✍ التمـريـن 38.04:

لتكن لدينا المعطيات العددية الحقيقية التالية:

$$nom = MENED$$
$$prenom = SIHEM$$
$$d_n = 17/03/1989$$
$$adr = TIARET$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـريـن 39.04:

لتكن لدينا معطيات التمرين 38.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال قيم المتغيرات أعلاه عن طريق لوحة المفاتيح. شريطة أن عملية الإدخال تتم كما هو موضح على الشاشة أدناه:

الشاشة

NOM :
PRENOM :
DATE DE NAISSANCE :
ADRESSE :

✍ التمـريـن 40.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  al, x , m , j , p , s , t  : string ;
begin
      al := 'ALGERIE' ;
      m := 'MOdule' ;
      x := 'TIARET' ;
      p := '0.12' ;
      j := 'admis' ;
      s  := 'AJOURNE' ;
      t  := ':=àç=56?./%µ' ;
end.
```

- **المطلـوب:**

1. حدد ماذا يقوم البرنامج أعلاه.

✍ التمـريــن 41.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  x , y , xy , v , w : string ;
begin
      a = '       ' ;
      y := 'DE  LA  PART  DE '
      v := '345.67' ;
      w := 2340.99 ;
      xy = 'UNIVERSITE' ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.

2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريــن 42.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  v1 , v5 , c , g , h : real ;
begin
      v := 1.230 ;
      v1 := 4.5 ;
      h = 66.7 ;
      g := 456.78
      s  := 450.99 ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريـن 43.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  v1 , v5 , c , v , g , h : real ;
begin
      v := 12.30 ;
      v1 := 45 ;
      h := 667 ;
      g := 45.678 ;
      s  := 450.78 ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.

✍ التمـريـن 44.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Clavier ;
var  a , b , c , d , e , f : string ;
begin
      readln ( a ) ;
      readln ( b ) ;
      readln ( c ) ;
      readln ( d ) ;
      readln ( e ) ;
end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. أكتب البرنامج الصحيح للبرنامج أعلاه.
3. ماذا يقوم البرنامج المتحصل عليه في السؤال 2 .

✍ التمـريـن 45.04:

لتكن المعطيات التالية:

$$FAR = 'hamid'$$
$$ZT = 'ali'$$
$$CF = 'mokhtar'$$
$$F = 'ahmed'$$
$$F = 'ahmed'$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج.
2. أكتب البرنامج الذي يسمح بإدخال قيم المتغيرات أعلاه عن طريق لوحة المفاتيح.

✍ التمـريـن 46.04:

لتكن المعطيات التالية:

$$F = sougueur$$
$$F1 = frenda$$
$$F2 = sonatiba$$
$$F3 = regina$$
$$F4 = eplf$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج.
2. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح.

✍ التمـريـن 47.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  e , r2 , r21 , X , L : string ;
```

```
begin
    L := 'aert' ;
    L := 'tyuiop' ;
    e := 'rte' ;
    r2 := 'nhgk' ;
    X := 'iutjgjn' ;
    e := '2500.789' ;
    r21 := 'ok,nbk' ;
    r2 := 'uti)àç_' ;
end.
```

- **المطلـوب:**

1. حدد القيمة النهائية لكل متغيرة.

✍ التمـريـن 48.04:

لتكن المعطيات التالية:

$$RTV = 'A67RTYUO89X'$$
$$NRT23 = 'H789AMI'$$
$$AZDFT = '23E4UIO89UU'$$
$$HJC41 = 'UI45JH4'$$
$$C23223 = '65UI4444OJK9'$$
$$XYZWW = '07100945'$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج.

2. أكتب البرنامج الذي يسمح بإدخال قيم المتغيرات أعلاه عن طريق لوحة المفاتيح.

✍ التمـريـن 49.04:

لتكن المعطيات التالية:

$$V = 'ARTYUOX'$$
$$NRT = 'HAMI'$$
$$DFT = 'E4UIO89'$$

$C41 = 'UIJH4'$

$C232 = '65UIOJK9'$

$W = '10945'$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج.

2. أكتب البرنامج الذي يسمح بإدخال قيم المتغيرات أعلاه عن طريق لوحة المفاتيح.

✍ التمـريـن 50.04:

ليكن البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  nom , t , h5 , yx , h9 , gh : string ;
begin
    xy := 'HaMiD' ;
    h5 := 'hai100.238' ;
    t := '0.34567' ;
    h9 := '4A27OHJtyuio1' ;
    nom := '10à)=azuio2008' ;
    t := 'Janvier,' ;
    h5  := 'Tiaret 14' ;
    gh  := '    tyiop    14      ' ;
end.
```

- **المطلـوب:**

1. بماذا يقوم البرنامج أعلاه.

✍ التمـريـن 51.04:

1. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات من النوع حرف عن طريق البرنامج.

2. قدم طريقة الكتابة (La Syntaxe) لإدخال المعطيات من النوع حرف عن طريق لوحة المفاتيح.

✍ التمـريـن 52.04:

لتكن المعطيات من النوع سلسلة أحرف التالية:

L , c , s , t , 7 , 6

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج (Introduction Des Données Par Programme) أي عن طريق التوجيه المباشر (Par l'Affectation Directe) .

✍ التمـريـن 53.04:

لتكن لدينا معطيات التمرين 52.04 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح.

✍ التمـريـن 54.04:

لتكن المعطيات التالية:

$$C2 = 'A'$$
$$VN1 = 'W'$$
$$D1F = 'G'$$
$$C21 = '0'$$
$$C23 = '8'$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج.

2. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح.

✍ التمـريـن 55.04:

حدد ماذا يقوم البرنامج التالي:

```
program Introduction_Des_Donnees_Par_Programme ;
var  alg , alg1 , ca , g , z1 : char ;
begin
    z1 := 'X' ;
    alg1 := 'x' ;
    alg := 'd' ;
    ca := '4' ;
```

```
g := ‘g’ ;
ca := ’h’;
alg  := ‘5’ ;
end.
```

✍ التمـريــن 56.04:

لتكن المعطيات التالية:

$$C = 'X'$$
$$N1 = 'H'$$
$$DF = 'E'$$
$$C1 = '4'$$
$$C2 = '9'$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق البرنامج.
2. أكتب البرنامج الذي يسمح بإدخال المعطيات أعلاه عن طريق لوحة المفاتيح.

الفصـــل الخامـــس

معـــالجـــة المعطيـــات العـــددية

Traitement Des Données Numériques

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- ماذا نعني بمعالجة المعطيات العددية.
- مختلف العمليات الحسابية التي تطبق على المعطيات العددية.
  1. عملية الجمع Opération d'Addition
  2. عملية الطرح Opération De Soustraction
  3. عملية الضرب (الجداء) Opération De Multiplication
  4. عملية القسمة Opération De Division
  5. عملية الرفع لقوة (الأس) Opération d'Exponentiation
- مخالف العلامات الحسابية المستخدمة في معالجة المعطيات العددية.
- مختلف الدوال التي تطبق على المعطيات العددية.
  1. الدالة **SQRT( )**
  2. الدالة **SQR( )**
  3. الدالة **ABS( )**
  4. الدالة **LN( )**
  5. الدالة **INT( )**
  6. الدالة **FRAC( )**

01.05 – تمهيـد: Introduction

بعد عمليتي التصريح بالمعطيات و إدخالها، يأتي دور العملية الثالثة الممثلة في **معالجـة المعطيـات** (Traitement Des Données) بجميع أنواعها (عددية و سلسلة أحرف). سوف يقتصر هذا الفصل على كيفية **معالجـة المعطيـات العدديـة** (Traitement Des Données Numériques) أي الصحيحة و الحقيقية على أن يتم التطرق إلى كيفية معالجة المعطيات من النوع سلسلة أحرف في الفصل الموالي.

02.05 – معالجـة المعطيـات العدديـة: Traitement Des Données Numériques

نعني بعملية معالجة المعطيات (المخزنة على مستوى أو في المتغيرات)، القيام بإجراء العمليات الحسابية (مثل الجمع، الطرح، الجداء، القسمة. . . .الخ) و العمليات العلاقاتية (مثل الأكبر، الأصغر، تساوي، أكبر أو يساوي. . . .الخ) و العمليات المنطقية بغرض الحصول معطيات جديدة تسمى **نتائـج** (Les Résultats) و التي سوف يتم توجيهها و تخزينها في متغيرات أخرى على أن يتم نشرها على الشاشة (Ecran) أو الطابعة (Imprimante).

03.05 – العمليـات الحسابيـة و العلامـات الحسابيـة:

Les Opérations Arithmétiques et Les Opérateurs Arithmétiques :

تتمثل العمليات الحسابية فيما يلي:

1. عملية الجمع Opération d'Addition
2. عملية الطرح Opération De Soustraction
3. عملية الضرب (الجداء) Opération De Multiplication
4. عملية القسمة الحقيقية Opération De Division Réelle
5. عملية القسمة الصحيحة Opération De Division Entière
6. عملية الرفع لقوة (الأس) Opération d'Exponentiation

سوف يتم التطرق بالتفضيل إلى كل عملية على حدى.

01.03.05 – عمليـة الجمـع و العلامـة + :

في بعض المرات يتم إدخال المعطيات العددية بغرض إنجاز أو القيام بعملية الجمع. العلامة التي تسمح بذالك تسمى **علامـة الجمـع** (Opérateur d'Addition) و التي يرمز لها بالرمز + . يتم استخدام علامة الجمع + وفقا لطريقة الكتابة التالية Suivant La Syntaxe Suivante:

Variable Numérique_3 := Variable Numérique_1 + Variable Numérique_2 ;

حيث أن:

- Variable Numérique_1 و Variable Numérique_2: تمثلان المتغيرة العددية الأولى و المتغيرة العددية الثانية التي نريد **جمـع** محتواهما بواسطة علامة الجمع +.
- + : تمثل علامة الجمع التي تقوم **بجمـع** محتوى المتغيرة العددية الأولى Variable Numérique_1 مع محتوى المتغيرة العددية الثانية Variable Numérique_2.
- Variable Numérique_3: تمثل متغيرة عددية ثالثة يتم فيها تخزين نتيجة عملية جمع محتوى المتغيرة العددية الأولى Variable Numérique_1 مع محتوى المتغيرة العددية الثانية Variable Numérique_2 بواسطة علامة الجمع +.
- **:=** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية الجمع إلى المتغيرة العددية الثالثة Variable Numérique_3.

بغرض توضيح كيفية إنجاز و القيام بعملية الجمع باستخدام علامة الجمع + نأخذ المثال التالي:

مثـــال 01.05:

لتكن لدينا المعطيات العددية من النوع الصحيح التالية:

4500 و 1250

يتم الوصول إلى البرنامج الذي يسمح بجمع المعطيتين أعلاه باستخدام علامة الجمع + كما يلي:

1. نقوم بالتصريح بمتغيرتين من النوع الصحيح (لأن المعطيتين أعلاه من النوع الصحيح) هما a و b ليتم تخزين فيهما المعطيتين على التوالي 4500 و 1250. عملية التصريح تتم كما يلي:

Var a , b : integer ;

2. نقوم بتخزين المعطيتين 4500 و 1250 في المتغيرتين a و b على التوالي. عملية التخزين تتم كما يلي:

a := 4500 ;
b := 1250 ;

3. القيام بعملية الجمع بين محتوى المتغيرة a و محتوى المتغيرة b بواسطة علامة الجمع +. يتم ذلك كما يلي: $a+b$
4. توجيه و تخزين نتيجة عملية الجمع في متغيرة ثالثة من النوع الصحيح و لتكن c. تتم عملية التوجيه و التخزين بواسطة علامة **:=** كما يلي:

c := a + b ;

5. نشر نتيجة عملية الجمع المخزنة في المتغيرة *c*. تتم عملية النشر بواسطة التعليمة *write* أو *writeln* كما يلي:

Write (c) ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program ADDITION ;
Var  a , b , c : integer ;
begin
    a:=4500 ;
    b:= 1250 ;
    c:= a + b ;
    write (c);
    readln;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
5750
```

ملاحظـة 01.05 :

مثلما يمكن لعلامة الجمع + القيام بجمع محتوى متغيرتين عدديتين، يمكن لها القيام بعملية جمع أكثر من محتوى متغيرتين عدديتين، أي جمع محتوى العديد من المتغيرات العددية. يتم ذلك وفقا لطريقة الكتابة التالية:

```
Var_ Num:=Var_Num_1+Var_ Num_2+Var_Num_3+…..+Var_Num_n ;
```

حيث أن:

- Var_Num_1,Var_ Num_2,Var_Num_3,…..,Var_Num_n: تمثل المتغيرات العددية التي نريد **جمـع** محتواهما.
- Var_ Num: تمثل متغيرة عددية يتم فيها تخزين نتيجة عملية جمع محتوى المتغيرات Var_Num_1,Var_ Num_2,Var_Num_3,…..,Var_Num_n بواسطة علامة الجمع +.

بغرض توضيح كيفية إنجاز و القيام بعملية جمع محتوى العديد من المتغيرات باستخدام علامة الجمع + نأخذ المثال التالي:

مثـــال 02.05:

لتكن لدينا المعطيات العددية الحقيقية و الصحيحة التالية

205.15 , 875 , 200.34 , 105.45 , 556 , 400 , 500.67

نريد كتابة البرنامج الذي يسمح بإنجاز و إجراء عمليات الجمع التالية:

1. $875+400$ و تخزين نتيجة الجمع في المتغيرة $r26$.
2. $875+556+400$ و تخزين النتيجة في $r236$.
3. $200.34+205.15$ و تخزين النتيجة في $r57$.
4. $205.15+200.34+105.45+500.67$ و تخزين النتيجة في $r1457$.
5. $400+500.67$ و تخزين النتيجة في $r12$.
6. جمع جميع القيم و تخزين النتيجة في r.

حتى نتمكن من كتابة البرنامج الذي يسمح بإجراء عمليات الجمع أعلاه نقوم بما يلي:

1. التصريح بـ 7 متغيرات (لأنه لدينا 7 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:
   - 3 متغيرات صحيحة و لتكن $e1,e2,e3$ لتخزين المعطيات الصحيحة 875 , 556 , 400. يتم ذلك كما يلي:

$$Var\ \ e1\ ,\ e2\ ,\ e3\ :\ integer\ ;$$

   - 4 متغيرات حقيقية و لتكن $r1,r2,r3,r4$ لتخزين المعطيات الحقيقية , 105.45 , 500.67 205.15 , 200.34 . يتم ذلك كما يلي:

$$Var\ \ r1\ ,\ r2\ ,\ r3\ ,\ r4\ \ :\ real\ ;$$

2. التصريح بـ 6 متغيرات (لأنه لدينا 6 عمليات جمع) بغرض تخزين نتيجة عمليات الجمع. هذه المتغيرات موزعة كما يلي:
   - متغيرة صحيحة و لتكن $r26$ لتخزين نتيجة عملية الجمع $875+400$.

- متغيرة صحيحة و لتكن $r236$ لتخزين نتيجة عملية الجمع $875+556+400$.
- متغيرة حقيقية و لتكن $r57$ لتخزين نتيجة عملية الجمع $205.15+200.34$.
- متغيرة حقيقية و لتكن $r1457$ لتخزين نتيجة عملية الجمع $205.15+200.34+105.45+500.67$.
- متغيرة حقيقية و لتكن $r12$ لتخزين نتيجة عملية الجمع $400+500.67$.
- متغيرة حقيقية و لتكن r لتخزين نتيجة جمع جميع المتغيرات.

تتم عملية التصريح بـ 6 متغيرات كما يلي:

```
Var  r26 , r236  : integer ;
Var  r57 , r1457 , r12 , r  : real ;
```

3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

```
r1 := 500.67 ;
r2 := 105.45 ;
r3 := 200.34 ;
r4 := 205.15 ;
e1 := 400 ;
e2 := 556 ;
e3 := 875 ;
```

4. القيام بالعمليات الحسابية أعلاه و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r26 := e1 + e3 ;
r236 := e1 + e2 + e3 ;
r57 : = r3 + r4 ;
r1457 : = r1 + r2 + r3 + r4 ;
r12 : = r1 + e1 ;
r : = r1 + e1 + e2 + r2 + r3 + e3 + r4 ;
```

5. نشر نتائج عمليات الجمع و المخزنة في المتغيرات *r26* ، *r236* ، *r57* ، *r1457* ، *r12* ، *r*. يتم ذلك كما يلي:

```
Writeln ( r26 ) ;
Writeln ( r236 ) ;
Writeln ( r57 ) ;
```

```
Writeln ( r1457 ) ;
Writeln ( r12 ) ;
Writeln ( r ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program ADDITION ;
Var  e1 , e2 , e3 , r26 , r236 : integer ;
Var  r1 , r2 , r3 , r4 , r57 , r1457 , r12 , r : real ;
Begin

    r1 := 500.67 ;
    r2 := 105.45 ;
    r3 := 200.34 ;
    r4 := 205.15 ;
    e1 := 400 ;
    e2 := 556 ;
    e3 := 875 ;
    r26 := e1 + e3 ;
    r236 := e1 + e2 + e3 ;
    r57 := r3 + r4 ;
    r1457 := r1 + r2 + r3 + r4 ;
    r12 := r1 + e1 ;
    r := r1 + e1 + e2 + r2 + r3 + e3 + r4 ;
    Writeln ( r26 ) ;
    Writeln ( r236 ) ;
    Writeln ( r57 ) ;
    Writeln ( r1457 ) ;
    Writeln ( r12 ) ;
    Writeln ( r ) ;
    readln;
end.
```

ملاحظـة 02.05 :

مثلما يمكن لعلامة الجمع + القيام بجمع محتوى المتغيرات العددية، يمكن لها كذلك حساب **مجموعة أو العديد من المعطيات** من النوع العددي دون اللجوء إلى إدخال هذه المعطيات في متغيرات ثم نقوم بعد ذلك بحساب مجموعها. يتم ذالك حسب طريقة الكتابة التالية:

Variable Numérique **:=** DN_1+ DN_2 + DN_3 + ….+ DN_n **;**

حيث أن:

- DN_1,DN_2,DN_3, ….,DN_n: تمثل المعطيات العددية التي نريد جمعها. قد يكون جميعها من النوع الصحيح أو من النوع الحقيقي و قد يكون البعض منها من النوع الصحيح و البعض الآخر من النوع الحقيقي.
- Variable Numérique: تمثل المتغيرة العددية التي يتم فيها تخزين عملية جمع المعطيات العددية. هذه المتغير تكون من النوع الصحيح إذا كانت جميع المعطيات من النوع الصحيح و تكون من النوع الحقيقي إذا كانت جميع المعطيات أو البعض منها (حتى و لو واحدة) من النوع الحقيقي.

بغرض التوضيح نأخذ المثال التالي:

مثـــال 03.05:

لتكن المعطيات الصحيحة و الحقيقية التالية

120 , 1500 , 4500 , 100.45 , 45.578 , 450.234

نريد إجراء عمليات الجمع التالية:

1. $120+1500+4500$
2. $100.45+45.578+450.234$
3. $120+1500+4500+450.234$
4. $120+1500+4500+100.45+45.578+450.234$

بغرض كتابة البرنامج الذي يسمح بانجاز عمليات الجمع أعلاه نتبع الخطوات التالية:

1. لا يتم التصريح بمتغيرات بغرض تخزين المعطيات العددية أعلاه
2. التصريح 4 متغيرات و لتكن $res_1, res_2, res_3, res_4$ بغرض تخزين نتائج عمليات الجمع أعلاه. هذه المتغيرات موزعة كما يلي:

- التصريح بالمتغيرة res_1 على أنها متغيرة من النوع الصحيح يغرض تخزين نتيجة العملية 1. يتم ذلك كما يلي:

```
Var  res_1 : integer ;
```

- التصريح بالمتغيرة res_2 على أنها متغيرة من النوع الحقيقي يغرض تخزين نتيجة العملية 2
- التصريح بالمتغيرة res_3 على أنها متغيرة من النوع الحقيقي يغرض تخزين نتيجة العملية 3
- التصريح بالمتغيرة res_4 على أنها متغيرة من النوع الحقيقي يغرض تخزين نتيجة العملية 4

```
Var  res_2 , res_3 , res_4  : real ;
```

3. القيام بعمليات الجمع و تخزين النتائج في المتغيرات التي تم التصريح بها في الخطوة الثانية

```
res_1 := 120 + 1500 + 4500 ;
res_2 := 100.45 + 45.578 + 450.234 ;
res_3 := 120 + 1500 + 4500 + 450.234 ;
res_4 := 120 + 1500 + 4500 + 100.45 + 45.578 + 450.234 ;
```

4. نشر نتائج عمليات الجمع و المخزنة في المتغيرات $res_1, res_2, res_3, res_4$

```
Writeln (res_1) ;
Writeln (res_2) ;
Writeln (res_3) ;
Writeln (res_4) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program ADDITION ;
Var  res_1 : integer ;
Var  res_2 , res_3 , res_4  : real ;
begin
    res_1 := 120 + 1500 + 4500 ;
    res_2 := 100.45 + 45.578 + 450.234 ;
    res_3 := 120 + 1500 + 4500 + 450.234 ;
    res_4 := 120 + 1500 + 4500 + 100.45 + 45.578 + 450.234 ;
    Writeln (res_1) ;
    Writeln (res_2) ;
    Writeln (res_3) ;
    Writeln (res_4) ;
```

```
readln;
end.
```

02.03.05 – عمليـة الطـرح و العلامـة – :

إضافة إلى عملية الجمع هناك عملية أخرى تسمى بعملية الطرح. العلامة التي تسمح بذالك تسمى **علامـة الطـرح** (Opérateur De Soustraction) و التي يرمز لها بالرمز – . يتم استخدام علامة الطرح – وفقا لطريقة الكتابة التالية:

```
Variable Numérique_3 := Variable Numérique_1 – Variable Numérique_2 ;
```

حيث أن:

- Variable Numérique_1 و Variable Numérique_2: تمثلان المتغيرة العددية الأولى و المتغيرة العددية الثانية التي نريد **طـرح** محتواهما بواسطة علامة الطرح –.
- – : تمثل علامة الطرح التي تقوم **بطـرح** محتوى المتغيرة العددية الثانية Variable Numérique_2 من محتوى المتغيرة العددية الأولى Variable Numérique_1.
- Variable Numérique_3: تمثل متغيرة عددية ثالثة يتم فيها تخزين نتيجة عملية طرح محتوى المتغيرة العددية الثانية Variable Numérique_2 من محتوى المتغيرة العددية الأولة Variable Numérique_1 بواسطة علامة الطرح –.
- **=:** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية الطرح إلى المتغيرة العددية الثالثة Variable Numérique_3.

كما يمكن إجراء و القيام بعملية الطرح على محتوى العديد من المتغيرات العددية وفقا لطريقة الكتابة التالية:

```
Var_ Num:=Var_Num_1–Var_ Num_2–Var_Num_3–…..–Var_Num_n ;
```

بغرض توضيح كيفية إنجاز و القيام بعملية الطرح باستخدام علامة الطرح – نأخذ المثال التالي:

مثـــال 04.05:

لتكن لدينا المعطيات العددية الصحيحة و الحقيقية التالية:

123 , 450.34 , 1250 , 900.45

نريد كتابة البرنامج الذي يسمح بإنجاز و إجراء عمليات الطرح التالية:

1. $123 - 1250$ و تخزين نتيجة الطرح في المتغيرة $r31$.
2. $1250 - 123$ و تخزين النتيجة في المتغيرة $r13$.
3. $450.34 - 900.45$ و تخزين النتيجة في $r42$.
4. $900.45 - 1250$ و تخزين النتيجة في $r34$.
5. $900.45 - 1250 - 450.34 - 123$ و تخزين النتيجة في r.

حتى نتمكن من كتابة البرنامج الذي يسمح بإجراء عمليات الطرح أعلاه نقوم بما يلي:

1. التصريح بـ 4 متغيرات (لأنه لدينا 4 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:

- 2 متغيرات صحيحة و لتكن $v1, v3$ لتخزين المعطيات الصحيحة 1250 , 123. يتم ذلك كما يلي:

Var v1 , v3 : integer ;

- 2 متغيرات حقيقية و لتكن $v2, v4$ لتخزين المعطيات الحقيقية 900.45 , 450.34 . يتم ذلك كما يلي:

Var v2 , v4 : real ;

2. التصريح بـ 5 متغيرات (لأنه لدينا 5 عمليات طرح) بغرض تخزين نتيجة عمليات الطرح. هذه المتغيرات موزعة كما يلي:

- متغيرة صحيحة و لتكن $r31$ لتخزين نتيجة عملية الطرح $123 - 1250$.
- متغيرة صحيحة و لتكن $r13$ لتخزين نتيجة عملية الطرح $1250 - 123$.
- متغيرة حقيقية و لتكن $r42$ لتخزين نتيجة عملية الطرح $450.34 - 900.45$.
- متغيرة حقيقية و لتكن $r34$ لتخزين نتيجة عملية الطرح $900.45 - 1250$.
- متغيرة حقيقية و لتكن r لتخزين نتيجة عملية الطرح $900.45 - 1250 - 450.34 - 123$.

تتم عملية التصريح بـ 6 متغيرات كما يلي:

Var r31 , r13 : integer ;

Var r42 , r34 , r : real ;

3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

v1 := 123 ;

v2 := 450.34 ;

v3 := 1250 ;

v4 := 900.45 ;

4. القيام بالعمليات الحسابية أعلاه و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r31 := v3 – v1 ;
r13 := v1 – v3  ;
r42 : = v4 – v2 ;
r34 : = v3 – v4 ;
r : = v1 – v2 – v3 – v4 ;
```

6. نشر نتائج عمليات الجمع و المخزنة في المتغيرات *r26* ، *r236* ، *r57* ، *r1457* ، *r12* ، *r*. يتم ذلك كما يلي:

```
Writeln ( r31 ) ;
Writeln ( r13 ) ;
Writeln ( r42 ) ;
Writeln ( r34 ) ;
Writeln ( r ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program SOUSTRACTION ;
Var  v1 , v3 , r31 , r13 : integer ;
Var  v2 , v4 ,  r42 , r34 , r : real ;
begin
    v1 := 123 ;
    v2 := 450.34 ;
    v3 := 1250 ;
    v4 := 900.45 ;
    r31 := v3 - v1 ;
    r13 := v1 - v3  ;
    r42 := v4 - v2 ;
    r34 := v3 - v4 ;
    r := v1 - v2 - v3 - v4 ;
    Writeln ( r31 ) ;
    Writeln ( r13 ) ;
    Writeln ( r42 ) ;
    Writeln ( r34 ) ;
    Writeln ( r ) ;
    readln;
end.
```

ملاحظـة 03.05 :

مثلما يمكن لعلامة الطرح – القيام بطرح محتوى المتغيرات العددية، يمكن لها كذلك طرح **مجموعة أو العديد من المعطيات** من النوع العددي دون اللجوء إلى إدخال هذه المعطيات في متغيرات ثم نقوم بعد ذلك بعملية الطرح. يتم ذالك حسب طريقة الكتابة التالية:

Variable Numérique **:=** DN_1− DN_2 − DN_3 − ….− DN_n **;**

حيث أن:

- DN_1,DN_2,DN_3, ….,DN_n: تمثل المعطيات العددية التي نريد طرحها. قد يكون جميعها من النوع الصحيح أو من النوع الحقيقي و قد يكون البعض منها من النوع الصحيح و البعض الآخر من النوع الحقيقي.
- Variable Numérique: تمثل المتغيرة العددية التي يتم فيها تخزين عملية طرح المعطيات العددية. هذه المتغير تكون من النوع الصحيح إذا كانت جميع المعطيات من النوع الصحيح و تكون من النوع الحقيقي إذا كانت جميع المعطيات أو البعض منها (حتى و لو واحدة) من النوع الحقيقي.

بغرض التوضيح نأخذ المثال التالي:

مثـــال 05.05:

لتكن المعطيات الصحيحة و الحقيقية التالية

450.234 , 45.578 , 100.45 , 4500 , 1500 , 120

نريد إجراء عمليات الطرح التالية:

5. $120-1500-4500$
6. $100.45-45.578-450.234$
7. $120-1500-4500-450.234$
8. $120-1500-4500-100.45-45.578-450.234$

بغرض كتابة البرنامج الذي يسمح بانجاز عمليات الطرح أعلاه نتبع الخطوات التالية:

1. لا يتم التصريح بمتغيرات بغرض تخزين المعطيات العددية أعلاه
2. التصريح 4 متغيرات و لتكن $res_1, res_2, res_3, res_4$ بغرض تخزين نتائج عمليات الجمع أعلاه. هذه المتغيرات موزعة كما يلي:

- التصريح بالمتغيرة res_1 على أنها متغيرة من النوع الصحيح يغرض تخزين نتيجة العملية 1. يتم ذلك كما يلي:

```
Var  res_1 : integer ;
```

- التصريح بالمتغيرة res_2 على أنها متغيرة من النوع الحقيقي يغرض تخزين نتيجة العملية 2
- التصريح بالمتغيرة res_3 على أنها متغيرة من النوع الحقيقي يغرض تخزين نتيجة العملية 3
- التصريح بالمتغيرة res_4 على أنها متغيرة من النوع الحقيقي يغرض تخزين نتيجة العملية 4

```
Var  res_2 , res_3 , res_4  : real ;
```

3. القيام بعمليات الطرح و تخزين النتائج في المتغيرات التي تم التصريح بها في الخطوة الثانية

```
res_1 := 120 − 1500 − 4500 ;
res_2 := 100.45 − 45.578 − 450.234 ;
res_3 := 120 − 1500 − 4500 − 450.234 ;
res_4 := 120 − 1500 − 4500 − 100.45 − 45.578 − 450.234 ;
```

4. نشر نتائج عمليات الطرح و المخزنة في المتغيرات $res_1, res_2, res_3, res_4$

```
Writeln (res_1) ;
Writeln (res_2) ;
Writeln (res_3) ;
Writeln (res_4) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program SOUSTRACTION ;
Var  res_1 : integer ;
Var  res_2 , res_3 , res_4  : real ;
begin
    res_1 := 120 - 1500 - 4500 ;
    res_2 := 100.45 - 45.578 - 450.234 ;
    res_3 := 120 - 1500 - 4500 - 450.234 ;
    res_4 := 120 - 1500 - 4500 - 100.45 - 45.578 - 450.234 ;
    Writeln (res_1) ;
    Writeln (res_2) ;
    Writeln (res_3) ;
```

```
    Writeln (res_4) ;
    readln;
end.
```

03.03.05 – عمليـة الضـرب (الجداء) و العلامـة *:

إضافة إلى عمليتي الجمع و الطرح هناك عملية أخرى تتمثل في عملية الجداء أو الضرب. العلامة التي تسمح بذالك تسمى **علامـة الضـرب** (Opérateur De Multiplication) و التي يرمز لها بالرمز * . يتم استخدام علامة الضرب * وفقا لطريقة الكتابة التالية:

Variable Numérique_3 **:=** Variable Numérique_1 * Variable Numérique_2 **;**

حيث أن:

- Variable Numérique_1 و Variable Numérique_2: تمثلان المتغيرة العددية الأولى و المتغيرة العددية الثانية التي نريد **ضـرب** محتواهما بواسطة علامة الضرب *.
- * : تمثل علامة الضرب التي تقوم **بضـرب** محتوى المتغيرة العددية الأولى Variable Numérique_1 بمحتوى المتغيرة العددية الثانية Variable Numérique_2.
- Variable Numérique_3: تمثل متغيرة عددية ثالثة يتم فيها تخزين نتيجة عملية ضرب محتوى المتغيرة العددية الأولى Variable Numérique_1 بمحتوى المتغيرة العددية الثانية Variable Numérique_2 بواسطة علامة الضرب *.
- **=:** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية الضـرب إلى المتغيرة العددية الثالثة Variable Numérique_3.

كما يمكن إجراء و القيام بعملية الضـرب على محتوى العديد من المتغيرات العددية وفقا لطريقة الكتابة التالية:

Var_ Num**:**=Var_Num_1*Var_ Num_2*Var_Num_3*…..*Var_Num_n **;**

بغرض توضيح كيفية إنجاز و القيام بعملية الضرب باستخدام علامة الضرب * نأخذ المثال التالي:

مثـــال 06.05:

لتكن لدينا المعطيات العددية الصحيحة و الحقيقية التالية:

15 , 3.75 , 15.05 , 4

نريد كتابة البرنامج الذي يسمح بإنجاز و إجراء عمليات الضرب التالية:

1. 4×15 و تخزين نتيجة الضرب في المتغيرة $r14$.
2. 15×3.75 و تخزين النتيجة في المتغيرة $r12$.
3. 4×15.05 و تخزين النتيجة في المتغيرة $r43$.
4. 3.75×15.05 و تخزين النتيجة في المتغيرة $r23$.
5. $4\times15.05\times3.75\times15$ و تخزين النتيجة في المتغيرة r.

حتى نتمكن من كتابة البرنامج الذي يسمح بإجراء عمليات الضرب أعلاه نقوم بما يلي:

1. التصريح بـ 4 متغيرات (لأنه لدينا 4 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:
   - 2 متغيرات صحيحة و لتكن $v1,v4$ لتخزين المعطيات الصحيحة 4 , 15. يتم ذلك كما يلي:

   Var v1 , v4 : integer ;
   - 2 متغيرات حقيقية و لتكن $v2,v3$ لتخزين المعطيات الحقيقية 15.05 , 3.75 . يتم ذلك كما يلي:

   Var v2 , v3 : real ;
2. التصريح بـ 5 متغيرات (لأنه لدينا 5 عمليات ضرب) بغرض تخزين نتيجة عمليات الضرب. هذه المتغيرات موزعة كما يلي:
   - متغيرة صحيحة $r14$ لتخزين نتيجة عملية الضرب 4×15.
   - متغيرة حقيقية $r12$ لتخزين نتيجة عملية الضرب 15×3.75.
   - متغيرة حقيقية $r43$ لتخزين نتيجة عملية الضرب 4×15.05.
   - متغيرة حقيقية $r23$ لتخزين نتيجة عملية الضرب 3.75×15.05.
   - متغيرة حقيقية r لتخزين نتيجة عملية الضرب $15\times3.75\times15.05\times4$.

   تتم عملية التصريح بـ 5 متغيرات كما يلي:

   Var r14 : integer ;

   Var r12 , r43 , r23 , r : real ;

في حالة الجداء يستحسن التصريح بجميع المتغيرات التي يتم فيها تخزين النتيجة عملية الضرب من النوع الحقيقي

3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

v1 := 15 ;

v2 := 3.75 ;

v3 := 15.05 ;

```
v4 := 4 ;
```

4. القيام بالعمليات الحسابية أعلاه و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r14 := v1 * v4 ;
r12 := v1 * v2 ;
r43 := v4 * v3 ;
r23 := v2 * v3 ;
r := v1 * v2 * v3 * v4 ;
```

5. نشر نتائج عمليات الضرب و المخزنة في المتغيرات *r14* ، *r12* ، *r43* ، *r23* ، *r*. يتم ذلك كما يلي:

```
Writeln ( r14 ) ;
Writeln ( r12 ) ;
Writeln ( r43 ) ;
Writeln ( r23 ) ;
Writeln ( r ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program MULTIPLICATION ;
Var  v1 , v4  : integer ;
Var  v2 , v3  : real ;
Var  r14   : integer ;
Var  r12 , r43 , r23 , r : real ;
begin
    v1 := 15 ;
    v2 := 3.75 ;
    v3 := 15.05 ;
    v4 := 4 ;
    r14 := v1 * v4 ;
    r12 := v1 * v2 ;
    r43 := v4 * v3 ;
    r23 := v2 * v3 ;
    r := v1 * v2 * v3 * v4 ;
    Writeln ( r14 ) ;
    Writeln ( r12 ) ;
    Writeln ( r43 ) ;
```

```
        Writeln ( r23 ) ;
        Writeln ( r ) ;
        readln;
    end.
```

ملاحظـة 04.05 :

مثلما يمكن لعلامة الضرب * القيام بضرب محتوى المتغيرات العددية، يمكن لها كذلك ضرب **مجموعة أو العديد من المعطيات** من النوع العددي دون اللجوء إلى إدخال هذه المعطيات في متغيرات ثم نقوم بعد ذلك بعملية الضرب يتم ذالك حسب طريقة الكتابة التالية:

Variable Numérique **:=** DN_1* DN_2 * DN_3 * ….* DN_n **;**

حيث أن:

- DN_1,DN_2,DN_3, ….,DN_n: تمثل المعطيات العددية التي نريد ضربها. قد يكون جميعها من النوع الصحيح أو من النوع الحقيقي و قد يكون البعض منها من النوع الصحيح و البعض الآخر من النوع الحقيقي.
- Variable Numérique: تمثل المتغيرة العددية التي يتم فيها تخزين عملية ضرب المعطيات العددية. هذه المتغيرة يفضل أن تكون من النوع الحقيقي.

بغرض التوضيح نأخذ المثال التالي:

مثـــال 07.05:

لتكن المعطيات الصحيحة و الحقيقية التالية

120 , 1500 , 4500 , 100.45 , 45.578 , 450.234

نريد إجراء عمليات الضرب التالية:

1. $120 \times 1500 \times 4500$
2. $100.45 \times 45.578 \times 450.234$
3. $120 \times 1500 \times 4500 \times 450.234$
4. $120 \times 1500 \times 4500 \times 100.45 \times 45.578 \times 450.234$

بغرض كتابة البرنامج الذي يسمح بانجاز عمليات الضرب أعلاه نتبع الخطوات التالية:

1. لا يتم التصريح بمتغيرات بغرض تخزين المعطيات العددية أعلاه

2. التصريح 4 متغيرات و لتكن $res_1, res_2, res_3, res_4$ من النوع الحقيقي بغرض تخزين نتائج عمليات الضرب أعلاه.

```
Var  res_1 , res_2 , res_3 , res_4  : real ;
```

3. القيام بعمليات الضرب و تخزين النتائج في المتغيرات التي تم التصريح بها في الخطوة الثانية

```
res_1 := 120 * 1500 * 4500 ;
res_2 := 100.45 * 45.578 * 450.234 ;
res_3 := 120 * 1500 * 4500 * 450.234 ;
res_4 := 120 * 1500 * 4500 * 100.45 * 45.578 * 450.234 ;
```

4. نشر نتائج عمليات الضرب و المخزنة في المتغيرات $res_1, res_2, res_3, res_4$

```
Writeln (res_1) ;
Writeln (res_2) ;
Writeln (res_3) ;
Writeln (res_4) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program MULTIPLICATION ;
Var  res_1 , res_2 , res_3 , res_4  : real ;
begin
    res_1 := 120 * 1500 * 4500 ;
    res_2 := 100.45 * 45.578 * 450.234 ;
    res_3 := 120 * 1500 * 4500 * 450.234 ;
    res_4 := 120 * 1500 * 4500 * 100.45 * 45.578 * 450.234 ;
    Writeln (res_1) ;
    Writeln (res_2) ;
    Writeln (res_3) ;
    Writeln (res_4) ;
readln;
end.
```

04.03.05 – عمليـة القسمـة الحقيقيـة و العلامـة / :

إضافة إلى عمليات الجمع و الطرح و الضرب التي يمكن القيام بها على المعطيات العددية، هناك عملية أخرى يمكن تطبيقها على هذا النوع من المعطيات. هذه العملية تتمثل في عملية القسمة. العلامة التي تسمح بذالك تسمى **علامـة القسمـة** (Opérateur De Division) و التي يرمز لها بالرمز / . يتم استخدام علامة القسمـة / وفقا لطريقة الكتابة التالية:

Variable Réelle := Variable Numérique_1 / Variable Numérique_2 ;

حيث أن:

- Variable Numérique_1: تمثل متغيرة عددية التي نريد قسمة محتواها. هذا المحتوى قد يكون من النوع الصحيح و قد يكون من النوع الحقيقي.
- Variable Numérique_2: تمثل متغيرة عددية التي نريد القسمة على محتواها. هذا المحتوى قد يكون من النوع الصحيح و قد يكون من النوع الحقيقي.
- / : تمثل علامة القسمة التي تقوم **بقسمة** محتوى المتغيرة العددية الأولى Variable Numérique_1 على محتوى المتغيرة العددية الثانية Variable Numérique_2. نشير إلى أن نتيجة عملية القسمة بالعلامة / عبارة عن **معطية من النوع الحقيقي** لذلك سوف يتم تخزينها في **متغيرة من النوع الحقيقي**.
- Variable Réelle: تمثل متغيرة عددية من النوع **الحقيقي** يتم فيها تخزين نتيجة عملية قسمة محتوى المتغيرة العددية الأولى Variable Numérique_1 على محتوى المتغيرة العددية الثانية Variable Numérique_2 بواسطة علامة القسمة /.
- =: : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية القسمة إلى المتغيرة العددية من النوع الحقيقي Variable Réelle.

ملاحظـة 05.05 :

- نتيجة عملية القسمة باستخدام العلامة / **دائما** عبارة عن معطية من النوع الحقيقي.
- المتغيرة التي نريد قسمة محتواها قد تكون من النوع الحقيقي و قد نكون من النوع الصحيح
- المتغيرة التي نريد القسمة على محتواها قد تكون من النوع الحقيقي و قد نكون من النوع الصحيح.

بغرض توضيح كيفية إنجاز و القيام بعملية القسمة باستخدام علامة القسمة / نأخذ المثال التالي:

مثـــال 08.05:

لتكن لدينا المعطيات العددية الصحيحة و الحقيقية التالية:

150 , 4 , 500.50 , 44.88

نريد كتابة البرنامج الذي يسمح بإنجاز و إجراء عمليات القسمة التالية:

1. $150 \div 4$ و تخزين نتيجة القسمة في المتغيرة $r12$.
2. $4 \div 150$ و تخزين النتيجة في المتغيرة $r21$.

3. $4 \div 44.88$ و تخزين النتيجة في المتغيرة $r42$.

4. $44.88 \div 500.50$ و تخزين النتيجة في المتغيرة $r34$.

5. $4 \div 500.50$ و تخزين النتيجة في المتغيرة $r32$.

حتى نتمكن من كتابة البرنامج الذي يسمح بإجراء عمليات القسمة أعلاه نقوم بما يلي:

1. التصريح بـ 4 متغيرات (لأنه لدينا 4 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:

- 2 متغيرات صحيحة و لتكن $v1, v2$ لتخزين المعطيات الصحيحة 4 , 150. يتم ذلك كما يلي:

```
Var  v1 , v2  : integer ;
```

- 2 متغيرات حقيقية و لتكن $v3, v4$ لتخزين المعطيات الحقيقية 44.88 , 500.50. يتم ذلك كما يلي:

```
Var  v3 , v4  : real ;
```

2. التصريح بـ 5 متغيرات (لأنه لدينا 5 عمليات قسمة) من النوع الحقيقي (و ليس من النوع الصحيح لأن نتيجة علامة القسمة / من النوع الحقيقي) بغرض تخزين نتيجة عمليات القسمة. هذه المتغيرات هي $r12, r21, r42, r34, r32$ تتم عملية التصريح كما يلي:

```
Var  r12 , r21 , r42 , r34 , r32  : real ;
```

3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

```
v1 := 150 ;
v2 := 4 ;
v3 := 500.50 ;
v4 := 44.88 ;
```

4. القيام بالعمليات الحسابية أعلاه و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r12 := v1 / v2 ;
r21 := v2 / v1 ;
r42 := v4 / v2 ;
r34 := v3 / v4 ;
r32 := v3 / v2  ;
```

5. نشر نتائج عمليات الضرب و المخزنة في المتغيرات $r12, r21, r42, r34, r32$. يتم ذلك كما يلي:

```
Writeln ( r12 ) ;
Writeln ( r21 ) ;
Writeln ( r42 ) ;
Writeln ( r34 ) ;
Writeln ( r32 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program DIVISION ;
Var  v1 , v2  : integer ;
Var  v3 , v4 , r12 , r21 , r42 , r34 , r32  : real ;
begin
    v1 := 150 ;
    v2 := 4 ;
    v3 := 500.50 ;
    v4 := 44.88 ;
    r12 := v1 / v2 ;
    r21 := v2 / v1 ;
    r42 := v4 / v2 ;
    r34 := v3 / v4 ;
    r32 := v3 / v2  ;
    Writeln ( r12 ) ;
    Writeln ( r21 ) ;
    Writeln ( r42 ) ;
    Writeln ( r34 ) ;
    Writeln ( r32 ) ;
    readln;
end.
```

05.03.05 – عمليـة القسمـة الصحيحـة و العلامـة DIV :

إضافة إلى عملية القسمة التي تم التطرق إليها في الفقرة السابقة و التي تطبق على أي نوع من المعطيات العددية، هناك عملية قسمة أخرى تطبق فقط على المعطيات العددية من النوع

الصحيح. العلامة التي تسمح بذالك تسمى **علامـة القسمـة الصحيحـة** (Opérateur De Division Entière) و التي يرمز لها بالرمز **DIV** . يتم استخدام علامة القسمة الصحيحة **DIV** وفقا لطريقة الكتابة التالية:

Var.Entière **:=** Var.Numérique_ Entière _1 **DIV** Var.Numérique_ Entière_2 **;**

حيث أن:

- Var.Numérique_ Entière _1: تمثل متغيرة عددية من النوع الصحيح التي نريد قسمة محتواها.
- Var.Numérique_ Entière_2: تمثل متغيرة عددية من النوع الصحيح التي نريد القسمة على محتواها.
- **DIV** : تمثل علامة القسمة الصحيحة التي تقوم **بقسمـة** محتوى المتغيرة العددية الصحيحة الأولى Var.Numérique_ Entière _1 على محتوى المتغيرة العددية الصحيحة الثانية Var.Numérique_ Entière_2. نشير إلى أن نتيجة عملية القسمة بالعلامة **DIV** عبارة عن **معطية من النوع الصحيح** لذلك سوف يتم تخزينها في **متغيرة من النوع الصحيح**.
- Var.Entière: تمثل متغيرة عددية من النوع **الصحيح** يتم فيها تخزين نتيجة عملية قسمة محتوى المتغيرة العددية الصحيحة الأولى Var.Numérique_ Entière _1 على محتوى المتغيرة العددية الصحيحة الثانية Var.Numérique_ Entière_2 بواسطة علامة القسمة الصحيحة **DIV**.
- **=:** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية القسمة إلى المتغيرة العددية من النوع الصحيح Var.Entière.

ملاحظـة 06.05 :

- نتيجة عملية القسمة باستخدام العلامة div عبارة عن معطية من النوع الصحيح.
- المتغيرة التي نريد قسمة محتواها باستخدام العلامة div يشترط أن تكون من النوع الصحيح.
- المتغيرة التي نريد القسمة على محتواها باستخدام العلامة div يشترط أن تكون من النوع الصحيح .
- إشارة نتيجة عملية القسمة موجبة إذا كان للمحتوى المتغيرتين نفس الإشارة

بغرض توضيح كيفية إنجاز و القيام بعملية القسمة باستخدام علامة القسمة الصحيحة div نأخذ الأمثلة التالية:

مثـــال 09.05:

1. نتيجة عملية القسمة الصحيحة **12 DIV 3** تساوي **4**
2. نتيجة عملية القسمة الصحيحة **13 DIV 3** تساوي **4**
3. نتيجة عملية القسمة الصحيحة **14 DIV 3** تساوي **4**
4. نتيجة عملية القسمة الصحيحة **16 DIV 4** تساوي **4**
5. نتيجة عملية القسمة الصحيحة **15 DIV 3** تساوي **5**
6. نتيجة عملية القسمة الصحيحة **14 DIV -3** تساوي **-4**
7. نتيجة عملية القسمة الصحيحة **-14 DIV -3** تساوي **4**
8. نتيجة عملية القسمة الصحيحة **-16 DIV 4** تساوي **4**

06.03.05 – باقي عمليـة القسمـة و العلامـة MOD :

العلامة MOD تسمح بتقديم باقي عملية القسمة، يتم ذالك وفقا لطريقة الكتابة التالية:

Var.Entière **:=** Var.Numérique_ Entière _1 **MOD** Var.Numérique_ Entière_2 **;**

حيث أن:

- Var.Numérique_ Entière _1: تمثل متغيرة عددية من النوع الصحيح التي نريد قسمة محتواها.
- Var.Numérique_ Entière_2: تمثل متغيرة عددية من النوع الصحيح التي نريد القسمة على محتواها.
- **MOD** : تمثل علامة باقي عملية قسمة محتوى المتغيرة العددية الصحيحة الأولى Var.Numérique_ Entière _1 على محتوى المتغيرة العددية الصحيحة الثانية Var.Numérique_ Entière_2. نشير إلى أن نتيجة عملية القسمة بالعلامة **MOD** عبارة عن **معطية من النوع الصحيح** لذلك سوف يتم تخزينها في **متغيرة من النوع الصحيح**.
- Var.Entière: تمثل متغيرة عددية من النوع **الصحيح** يتم فيها باقي عملية قسمة.
- **:=** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية القسمة إلى المتغيرة العددية من النوع الصحيح Var.Entière.

مثـــال 10.05:

1. **6 MOD 4** تساوي **4**
2. **4 MOD 6** تساوي **2**
3. **7 MOD 14** تساوي **0**
4. **7 MOD -14** تساوي **0**
5. **5 MOD -6** تساوي **-1**
6. **4 MOD -6** تساوي **-2**

07.03.05 – عمليـة الرفـع لقـوة (الأس) و العلامـة ** :

إضافة إلى العمليات المذكورة أعلاه، هناك عملية أخرى يمكن تطبيقها على المعطيات العددية بنوعيها الصحيح و الحقيقي. هذه العملية تتمثل في عملية **الرفع لقوة** أو عملية الأس. العلامة التي تسمح بذالك تسمى **علامـة الأس** (Opérateur D'exponentiation) و التي يرمز لها بالرمز **. يتم استخدام علامة القسمة ** وفقا لطريقة الكتابة التالية:

Variable Numérique **:=** Variable Numérique_1 ** Variable Numérique_2 **;**

حيث أن:

- Variable Numérique_1: تمثل المتغيرة العددية التي نريد أن نرفع محتواها لقوة. هذا المحتوى قد يكون من النوع الصحيح و قد يكون من النوع الحقيقي.
- Variable Numérique_2: تمثل المتغيرة العددية التي محتواها يمثل الأس. هذا المحتوى قد يكون من النوع الصحيح و قد يكون من النوع الحقيقي.
- ** : تمثل علامة **الأس** التي تقوم **برفـع** محتوى المتغيرة العددية الأولى Variable Numérique_1 **أس** محتوى المتغيرة العددية الثانية Variable Numérique_2.
- Variable Numérique: تمثل متغيرة عددية يتم فيها تخزين نتيجة عملية الأس. يستحسن أن تكون من النوع **الحقيقي**.
- **:=** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية **الأس** إلى المتغيرة العددية Variable Numérique.

بغرض توضيح كيفية إنجاز و القيام بعملية الضرب باستخدام علامة الضرب * نأخذ المثال التالي:

مثـــال 11.05:

لتكن لدينا المعطيات العددية الصحيحة و الحقيقية التالية:

5 , 0.5 , 5.4 , 3

نريد كتابة البرنامج الذي يسمح بإنجاز و إجراء عمليات القسمة التالية:

1. $5^{0.5}$ و تخزين نتيجة الضرب في المتغيرة $r12$.
2. $(0.5)^{3}$ و تخزين نتيجة الضرب في المتغيرة $r24$.
3. $(5.4)^{5}$ و تخزين نتيجة الضرب في المتغيرة $r31$.
4. $(0.5)^{5.4}$ و تخزين نتيجة الضرب في المتغيرة $r23$.
5. $(0.5+5.4)^{5}$ و تخزين نتيجة الضرب في المتغيرة rr.
6. $(5+0.5+3)^{5.4}$ و تخزين نتيجة الضرب في المتغيرة r.

حتى نتمكن من كتابة البرنامج الذي يسمح بإجراء عمليات القسمة أعلاه نقوم بما يلي:

1. التصريح بـ 4 متغيرات (لأنه لدينا 4 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:
   - 2 متغيرات صحيحة و لتكن $v1,v4$ لتخزين المعطيات الصحيحة 3 , 5. يتم ذلك كما يلي:

   Var v1 , v4 : integer ;
   - 2 متغيرات حقيقية و لتكن $v2,v3$ لتخزين المعطيات الحقيقية 5.4 , 0.5. يتم ذلك كما يلي:

   Var v2 , v3 : real ;
2. التصريح بـ 6 متغيرات (لأنه لدينا 6 عمليات) من النوع الحقيقي بغرض تخزين نتيجة عمليات القسمة. هذه المتغيرات هي $r12,r24,r31,r23,rr,r$ تتم عملية التصريح كما يلي:

   Var r12 , r24 , r31 , r23 , rr , r : real ;
3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

   v1 := 5 ;
   v2 := 0.5 ;
   v3 := 5.4 ;
   v4 := 3 ;
4. القيام بالعمليات الحسابية أعلاه و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

   *r12 := v1** v2 ;*
   *r24 := v2 ** v4 ;*
   *r31 := v3 ** v1 ;*
   *r23 := v2 ** v3 ; ;*

```
rr := ( v2 + v3 ) ** v1  ;
r := ( v1 + v2 + v4 ) ** v3  ;
```

5. نشر نتائج عمليات الضرب و المخزنة في المتغيرات $r12, r24, r31, r23, rr, r$. يتم ذلك كما يلي:

```
Writeln ( r12 ) ;
Writeln ( r24 ) ;
Writeln ( r31 ) ;
Writeln ( r23 ) ;
Writeln ( rr ) ;
Writeln ( r ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program PUISSANCE ;
Var  v1, v4, v2 , v3 , r12 , r24 , r31 , r23 , rr , r  : real ;
begin
    v1 := 5 ;
    v2 := 0.5 ;
    v3 := 5.4 ;
    v4 := 3 ;
    r12 := v1** v2 ;
    r24 := v2 ** v4 ;
    r31 := v3 ** v1 ;
    r23 := v2 ** v3 ;  ;
    rr := ( v2 + v3 ) ** v1  ;
    r := ( v1 + v2 + v4 ) ** v3  ;
    Writeln ( r12 ) ;
    Writeln ( r24 ) ;
    Writeln ( r31 ) ;
    Writeln ( r23 ) ;
    Writeln ( rr ) ;
    Writeln ( r ) ;
    readln;
end.
```

04.05 – الدوال على المعطيـات العدديـة: Les Fonctions Sur Les Données Numériques

إذا أردنا حساب الجذر التربيعي لمحتوى متغيرة ما و لتكن a، أي نريد حساب $\sqrt[2]{a}$ فإننا نلجأ إلى حسابه باستخدام علامة الأس كما يلي:

$$\sqrt[2]{a} = (a)^{\frac{1}{2}} = (a)^{0.5}$$

يمكن حساب الجذر التربيعي دون اللجوء إلى علامة الأس و ذلك باستخدام الدالة **SQRT( )** التي تقوم بحساب الجذر التربيعي كما يلي:

$$SQRT(a)$$

كما يمكن حساب مربع معطية عددية و لتكن b ما دون اللجوء إلى علامة الأس و ذلك باستخدام الدالة **SQR( )** التي تقوم بحساب المربع كما يلي:

$$SQR(b)$$

من خلال هذه الفقرة سوف يتم التطرق إلى بعض الدوال التي تطبق على المعطيات العددية بغرض القيام بعمليات حسابية معينة. هذه الدوال هي:

1. الدالة **SQRT( )** : تسمح بحساب الجذر التربيعي
2. الدالة **SQR( )** : تسمح بحساب المربع
3. الدالة **ABS( )** : تسمح بحساب القيمة المطلقة
4. الدالة **LN( )** : تسمح بحساب اللوغريتم النبيري
5. الدالة **INT( )** : تسمح بحساب الجزء الصحيح
6. الدالة **FRAC( )** : تسمح بحساب الجزء الكسري

سوف يتم التطرق بالتفصيل إلى كيفية استخدام كل دالة

01.04.05 – الجذر التربيعي: الدالـة SQRT ( )

تقوم الدالة SQRT( ) بحساب الجذر التربيعي لمعطية من النوع العددي سواء كانت صحيحة أو حقيقية. يتم ذالك وفقا لطريقة الكتابة التالية:

```
Variable Réelle := SQRT( Variable Numérique ) ;
```

حيث أن:

- Variable Numérique: تمثل المتغيرة العددية التي نريد أن نحسب الجذر التربيعي لمحتواها. هذا المحتوى قد يكون من النوع الصحيح و قد يكون من النوع الحقيقي.

- **SQRT ( )** : تمثل الدالة التي نقوم بحساب الجذر التربيعي لمحتوى المتغيرة العددية Variable Numérique.
- Variable Réelle: تمثل متغيرة عددية من النوع الحقيقي يتم فيها تخزين نتيجة عملية حساب الجذر التربيعي.
- **:=** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية الجذر التربيعي إلى المتغيرة العددية من النوع الحقيقي Variable Réelle.

بغرض توضيح كيفية حساب الجذر التربيعي بواسطة الدالة SQRT( ) نأخذ المثال التالي:

مثـــال 12.05:

يتم حساب الجذر التربيعي للمتغيرتين العدديتين $a = 4096$ و $b = 51.84$ كما يلي:

- $SQRT(a)$
- $SQRT(b)$

مثـــال 13.05:

لتكن لدينا المعطيات العددية التالية:

15625 , 25 , 20.25 , 2550.25

بغرض كتابة البرنامج الذي يسمح بحساب الجذر التربيعي للمعطيات أعلاه:

1. التصريح بـ 4 متغيرات (لأنه لدينا 4 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:

- 2 متغيرات صحيحة و لتكن $v1, v2$ لتخزين المعطيات الصحيحة 25 , 15625. يتم ذلك كما يلي:

```
Var  v1 , v2  : integer ;
```

- 2 متغيرات حقيقية و لتكن $v3, v4$ لتخزين المعطيات الحقيقية 2550.25 , 20.25. يتم ذلك كما يلي:

```
Var  v3 , v4  : real ;
```

2. التصريح بـ 6 متغيرات (لأنه لدينا 6 عمليات) من النوع الحقيقي بغرض تخزين نتيجة عمليات حساب الجذر التربيعي. هذه المتغيرات هي $r1, r2, r3, r2$ تتم عملية التصريح كما يلي:

```
Var  r1 , r2 , r3 , r4  : real ;
```

3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

```
v1 := 15625 ;
v2 := 25 ;
```

```
v3 := 20.25 ;
v4 := 2550.25 ;
```

4. القيام بالعمليات الحسابية أعلاه و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r1 := SQRT( v1 ) ;
r2 := SQRT( v2 ) ;
r3 := SQRT( v3 ) ;
r4 := SQRT( v4 ) ;  ;
```

5. نشر نتائج عمليات الضرب و المخزنة في المتغيرات $r1, r2, r3, r2$. يتم ذلك كما يلي:

```
Writeln ( r1 ) ;
Writeln ( r2 ) ;
Writeln ( r3 ) ;
Writeln ( r4 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program RACINE_CAREE ;
Var  v1 , v2   : integer ;
Var  v3 , v4  : real ;
Var  r1 , r2 , r3 , r4  : real ;
begin
    v1 := 15625 ;
    v2 := 25 ;
    v3 := 20.25 ;
    v4 := 2550.25 ;
    r1 := SQRT( v1 ) ;
    r2 := SQRT( v2 ) ;
    r3 := SQRT( v3 ) ;
    r4 := SQRT( v4 ) ;
    Writeln ( r1 ) ;
    Writeln ( r2 ) ;
    Writeln ( r3 ) ;
    Writeln ( r4 ) ;
    readln;
end.
```

ملاحظـة 07.05 :

مثلما يمكن للدالة ( )SQRT حساب الجذر التربيعي **لمحتوى** متغيرة من النوع العددي، يمكن لها كذلك حساب الجذر التربيعي **لمعطية** من النوع العددي دون اللجوء إلى إدخال هذه المعطية في متغيرة ثم نقوم بعد ذلك حساب الجذر التربيعي لهذه المتغيرة. يتم ذالك حسب طريقة الكتابة التالية:

Variable Réelle **:= SQRT(** Donnée Numérique **) ;**

حيث أن:

- Donnée Numérique: تمثل المعطية العددية التي نريد أن نحسب الجذر التربيعي لها. هذه المعطية قد تكون من النوع الصحيح و قد تكون من النوع الحقيقي.

مثـــال 14.05:

يتم حساب الجذر التربيعي للمعطيتين العدديتين 4096 و 51.84 كما يلي:

- *SQRT*(4096)
- *SQRT*(51.84)

مثـــال 15.05:

لتكن لدينا معطيات المثال 13.05 أعلاه:

بغرض كتابة البرنامج الذي يسمح بحساب الجذر التربيعي لمعطيات المثال 13.05 نتبع الخطوات التالية:

1. لا يتم التصريح بأربع متغيرات بغرض تخزين المعطيات العددية.
2. التصريح بـ 4 متغيرات (لأنه لدينا 4 عمليات) من النوع الحقيقي بغرض تخزين نتيجة عمليات حساب الجذر التربيعي. هذه المتغيرات هي $r1, r2, r3, r2$ تتم عملية التصريح كما يلي:

Var r1 , r2 , r3 , r4 : real ;

3. لا يتم تخزين المعطيات في متغيرات.
4. القيام بالعمليات الحسابية أعلاه و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r1 := SQRT( 15625 ) ;
r2 := SQRT( 25 ) ;
r3 := SQRT( 20.25 ) ;
r4 := SQRT( 2550.25 ) ;
```

5. نشر نتائج عمليات الضرب و المخزنة في المتغيرات $r1, r2, r3, r2$. يتم ذلك كما يلي:

```
Writeln ( r1 ) ;
Writeln ( r2 ) ;
Writeln ( r3 ) ;
Writeln ( r4 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program RACINE_CAREE ;
Var  r1 , r2 , r3 , r4  : real ;
begin
      r1 := SQRT( 15625 ) ;
      r2 := SQRT( 25 ) ;
      r3 := SQRT( 20.25 ) ;
      r4 := SQRT( 2550.25 ) ;
      Writeln ( r1 ) ;
      Writeln ( r2 ) ;
      Writeln ( r3 ) ;
      Writeln ( r4 ) ;
readln;
end.
```

02.04.05 – التربيـع: الدالـة ( ) SQR

تقوم الدالة ( )SQR بحساب **مربـع** معطية عددية سواء كانت صحيحة أو حقيقية. يتم ذالك وفقا لطريقة الكتابة التالية:

Variable Numérique **:= SQR(** Var_Numérique **) ;**

حيث أن:

- Var_Numérique: تمثل المتغيرة العددية التي نريد أن نحسب مربع محتواها. هذا المحتوى قد يكون من النوع الصحيح و قد يكون من النوع الحقيقي.
- **SQR( )** : تمثل الدالة التي تقوم بحساب **مربـع** محتوى المتغيرة العددية Variable Numérique.
- Variable Numérique: تمثل متغيرة عددية يتم فيها تخزين نتيجة عملية حساب المربع. نوع المتغيرة Variable Numérique من نوع المتغيرة Var_Numérique أي تكون من النوع الصحيح إذا كانت المتغيرة Var_Numérique من النوع الصحيح أما إذا كانت

Var_Numérique من النوع الحقيقي فإن Variable Numérique تكون من النوع الحقيقي

- **=:** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية المربع إلى المتغيرة العددية Variable Numérique.

بغرض توضيح كيفية حساب المربع بواسطة الدالة SQR( ) نأخذ المثال التالي:

مثـــال 16.05:

يتم حساب مربع المعطيات العددية $a = 123$ و $b = 30.75$ و $c = 100.34$ كما يلي:

- $SQR(a)$
- $SQR(b)$
- $SQR(c)$

مثـــال 17.05:

لتكن لدينا المعطيات العددية الحقيقية و الصحيحة التالية:

45 , 1000 , 230.34 , 25.25

بغرض كتابة البرنامج الذي يسمح بحساب مربع المعطيات أعلاه يجب:

1. التصريح بـ 4 متغيرات (لأنه لدينا 4 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:

- 2 متغيرات صحيحة و لتكن $v1, v2$ لتخزين المعطيات الصحيحة 1000 , 45. يتم ذلك كما يلي:

Var v1 , v2 : integer ;

- 2 متغيرات حقيقية و لتكن $v3, v4$ لتخزين المعطيات الحقيقية 25.25 , 230.34. يتم ذلك كما يلي:

Var v3 , v4 : real ;

2. التصريح بـ 4 متغيرات (لأنه لدينا 4 عمليات) بغرض تخزين نتيجة عمليات حساب المربع. هذه المتغيرات هي $r1, r2, r3, r2$ موزعة كما يلي:

- 2 متغيرات صحيحة و لتكن $r1, r2$ لتخزين مربع القيم الصحيحة 1000 , 45. يتم ذلك كما يلي:

Var r1 , r2 : integer ;

يمكن التصريح بالمتغيرات الصحيحة $r1, r2$ على أنها متغيرات حقيقية دون أن يؤثر ذلك على البرنامج أي:

Var r1 , r2 : real ;

- 2 متغيرات حقيقية و لتكن $r3, r4$ لتخزين مربع القيم الحقيقية 25.25 , 203.34. يتم ذلك كما يلي:

Var r3 , r4 : real ;

3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

v1 := 45 ;

v2 := 1000 ;

v3 := 230.34 ;

v4 := 25.25 ;

4. القيام بعمليات حساب المربع و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

r1 := SQR(v1) ;

r2 := SQR(v2) ;

r3 := SQR(v3) ;

r4 := SQR(v4) ;

5. نشر نتائج عمليات المربع المخزنة في المتغيرات $r1, r2, r3, r2$. يتم ذلك كما يلي:

Writeln (r1) ;

Writeln (r2) ;

Writeln (r3) ;

Writeln (r4) ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program CARREE ;
Var  v1 , v2 , r1 , r2  : integer ;
Var  v3 , v4 , r3 , r4  : real ;
begin
    v1 := 45 ;
    v2 := 1000 ;
    v3 := 230.34 ;
    v4 := 25.25 ;
    r1 := SQR( v1 ) ;
    r2 := SQR( v2 ) ;
    r3 := SQR( v3 ) ;
```

```
    r4 := SQR( v4 ) ;
    Writeln ( r1 ) ;
    Writeln ( r2 ) ;
    Writeln ( r3 ) ;
    Writeln ( r4 ) ;
    readln;
end.
```

ملاحظـة 08.05 :

مثلما يمكن للدالة **( )SQR** حساب مربع **محتوى** متغيرة من النوع العددي، يمكن لها كذلك حساب مربع **معطيـة** من النوع العددي دون اللجوء إلى إدخال هذه المعطية في متغيرة ثم نقوم بعد ذلك حساب مربع هذه المتغيرة. يتم ذالك حسب طريقة الكتابة التالية:

Variable Numérique **:= SQR(** Donnée Numérique **) ;**

حيث أن:

- Donnée Numérique: تمثل المعطية العددية التي نريد أن نحسب مربعها. هذه المعطية قد تكون من النوع الصحيح و قد تكون من النوع الحقيقي.
- Variable Numérique: تمثل المتغيرة العددية التي يتم فيها تخزين نتيجة عملية المربع. هذه المتغيرة تكون من النوع الصحيح إذا كانت Donnée Numérique من النوع الصحيح أما إذا كانت Donnée Numérique من النوع الحقيقي فإن Variable Numérique تكون من النوع الحقيقي.

مثــال 18.05:

يتم حساب مربع القيم 40.123 و 145 و 3.005 كما يلي:

- *SQR*(40.123)
- *SQR*(145)
- *SQR*(3.005)

مثــال 19.05:

لتكن لدينا المعطيات العددية التالية:

782.09 , 10.32 , 367 , 450

بغرض كتابة البرنامج الذي يسمح بحساب مربع المعطيات أعلاه نتبع الخطوات التالية:

1. لا يتم التصريح بأربع متغيرات بغرض تخزين المعطيات العددية أعلاه.

2. التصريح بـ 4 متغيرات (لأنه لدينا 4 عمليات) بغرض تخزين نتيجة عمليات حساب المربع. هذه المتغيرات هي $r1, r2, r3, r2$ موزعة كما يلي:

- 2 متغيرات من النوع الصحيح لتخزين مربع المعطيات العددية الصحيحة 367 , 450 يتم ذلك كما يلي:

```
Var  r1 , r2 : integer ;
```

يمكن التصريح بالمتغيرات الصحيحة $r1, r2$ على أنها متغيرات حقيقية دون أن يؤثر ذلك على البرنامج أي:

```
Var  r1 , r2  : real ;
```

- 2 متغيرات من النوع الصحيح لتخزين مربع المعطيات العددية الصحيحة , 10.32 782.09 يتم ذلك كما يلي:

```
Var  r3 , r4  : real ;
```

3. لا يتم تخزين المعطيات في المتغيرات.

4. القيام بالعمليات الحسابية أعلاه و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r1 := SQR( 450 ) ;
r2 := SQR( 367 ) ;
r3 := SQR( 10.32 ) ;
r4 := SQR( 782.09 ) ;
```

5. نشر نتائج عمليات الضرب و المخزنة في المتغيرات $r1, r2, r3, r2$. يتم ذلك كما يلي:

```
Writeln ( r1 ) ;
Writeln ( r2 ) ;
Writeln ( r3 ) ;
Writeln ( r4 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program CARREE ;
Var  r1 , r2 : integer ;
Var  r3 , r4  : real ;
begin
    r1 := SQR( 450 ) ;
```

```
r2 := SQR( 367 ) ;
r3 := SQR( 10.32 ) ;
r4 := SQR( 782.09 ) ;
Writeln ( r1 ) ;
Writeln ( r2 ) ;
Writeln ( r3 ) ;
Writeln ( r4 ) ;
readln;
end.
```

03.04.05 – القيمـة المطلقـة: الدالـة ABS( )

تقوم الدالة ABS( ) بحساب **القيمـة المطلقـة** لمعطية عددية سواء كانت صحيحة أو حقيقية. يتم ذالك وفقا لطريقة الكتابة التالية:

Variable Numérique **:= ABS(** Var_Numérique **) ;**

حيث أن:

- Var_Numérique: تمثل المتغيرة العددية التي نريد أن نحسب القيمة المطلقة لمحتواها. هذه المتغيرة قد تكون من النوع الصحيح و قد تكون من النوع الحقيقي.
- **ABS( )** : تمثل الدالة التي تقوم بحساب **القيمـة المطلقـة** لمحتوى المتغيرة العددية Var_Numérique و تخزينه في المتغيرة العددية Variable Numérique.
- Variable Numérique: تمثل متغيرة عددية يتم فيها تخزين نتيجة عملية حساب **القيمـة المطلقـة**. المتغيرة Variable Numérique نوعها من نوع المتغيرة Var_Numérique بمعنى أن المتغيرة Variable Numérique من النوع الحقيقي إذا كانت المتغيرة Var_Numérique من النوع الحقيقي. أما إذا كانت هذه الأخيرة من النوع الصحيح فإن المتغيرة Variable Numérique يجب أن تكون من النوع الصحيح.
- **:=** : تمثل علامة تساوي تسمح أو تقوم بتخزين و تحويل و توجيه نتيجة عملية القيمة المطلقة إلى المتغيرة العددية Variable Numérique.

بغرض توضيح كيفية حساب **القيمـة المطلقـة** بواسطة الدالة ABS( ) نأخذ المثال التالي:

مثـــال 20.05:

يتم حساب مربع المعطيات العددية $x = -12.55$ و $y = 20.34$ و $z = -150$ و $w = 45$ كما يلي:

- $ABS(x)$ و التي تساوي 12.55
- $ABS(y)$ و التي تساوي 20.34
- $ABS(z)$ و التي تساوي 150
- $ABS(w)$ و التي تساوي 45

مثـــال 21.05:

لتكن لدينا المعطيات العددية الحقيقية و الصحيحة التالية:

$$1250 \ , \ -4250 \ , \ 56.09 \ , \ -423.567$$

بغرض كتابة البرنامج الذي يسمح بحساب القيمة المطلقة للمعطيات أعلاه نتبع الخطوات التالية:

1. التصريح بـ 4 متغيرات (لأنه لدينا 4 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:

- 2 متغيرات حقيقية و لتكن $v1, v2$ لتخزين المعطيات الحقيقية $-423.567, 56.09$. يتم ذلك كما يلي:

```
Var  v1 , v2  : real ;
```

- 2 متغيرات صحيحة و لتكن $v3, v4$ لتخزين المعطيات الصحيحة $-4250, 1250$. يتم ذلك كما يلي:

```
Var  v3 , v4  : integer ;
```

2. التصريح بـ 4 متغيرات (لأنه لدينا 4 عمليات) بغرض تخزين نتيجة عمليات حساب القيمة المطلقة. هذه المتغيرات هي $r1, r2, r3, r2$ موزعة كما يلي:

- 2 متغيرات حقيقية و لتكن $r1, r2$ لتخزين القيمة المطلقة للقيم الحقيقية $-423.567, 56.09$. يتم ذلك كما يلي:

```
Var  r1 , r2  : real ;
```

- 2 متغيرات صحيحة و لتكن $r3, r4$ لتخزين مربع القيم الصحيحة $-4250, 1250$. يتم ذلك كما يلي:

```
Var  r3 , r4  : integer ;
```

3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى:

```
v1 := - 423.567 ;
```

v2 := 56.09 ;

v3 := - 4250 ;

v4 := 1250 ;

4. القيام بعمليات حساب القيمة المطلقة و تخزين النتائج في المتغيرات الموافقة لكل نتيجة:

r1 := ABS(v1) ;

r2 := ABS(v2) ;

r3 := ABS(v3) ;

r4 := ABS(v4) ;

5. نشر نتائج عمليات حساب القيمة المطلقة و المخزنة في المتغيرات $r1, r2, r3, r2$:

Writeln (r1) ;

Writeln (r2) ;

Writeln (r3) ;

Writeln (r4) ;

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program VALEUR_ABSOLUT ;
Var  v1 , v2 , r1 , r2 : real ;
Var  v3 , v4 , r3 , r4  : integer ;
begin
    v1 := - 423.567 ;
    v2 := 56.09 ;
    v3 := - 4250 ;
    v4 := 1250 ;
    r1 := ABS( v1 ) ;
    r2 := ABS( v2 ) ;
    r3 := ABS( v3 ) ;
    r4 := ABS( v4 ) ;
    Writeln ( r1 ) ;
    Writeln ( r2 ) ;
    Writeln ( r3 ) ;
    Writeln ( r4 ) ;
    readln;
end.
```

ملاحظـة 09.05 :

مثلما يمكن للدالة **ABS( )** حساب القيمة المطلقة **لمحتوى** متغيرة من النوع العددي، يمكن لها كذلك حساب القيمة المطلقة **لمعطية** من النوع العددي دون اللجوء إلى إدخال هذه المعطية في متغيرة ثم نقوم بعد ذلك حساب القيمة المطلقة هذه المتغيرة. يتم ذالك حسب طريقة الكتابة التالية:

Variable Numérique **:= ABS(** Donnée Numérique **) ;**

حيث أن:

- Donnée Numérique: تمثل المعطية العددية التي نريد أن نحسب مربعها. هذه المعطية قد تكون من النوع الصحيح و قد تكون من النوع الحقيقي.
- Variable Numérique: تمثل المتغيرة العددية التي يتم فيها تخزين نتيجة عملية المربع. هذه المتغيرة تكون من النوع الصحيح إذا كانت Donnée Numérique من النوع الصحيح أما إذا كانت Donnée Numérique من النوع الحقيقي فإن Variable Numérique تكون من النوع الحقيقي.

مثـــال 22.05:

يتم حساب مربع القيم 40.123 – و 145– و 305.45 و 190 كما يلي:

- $ABS(-40.123)$ و التي تساوي 40.123
- $ABS(-145)$ و التي تساوي 145
- $ABS(305.45)$ و التي تساوي 305.45
- $ABS(190)$ و التي تساوي 190

مثـــال 23.05:

لتكن لدينا المعطيات العددية التالية:

$$-450 \;,\; 367 \;,\; 10.32 \;,\; -782.09$$

بغرض كتابة البرنامج الذي يسمح بحساب القيمة المطلقة للمعطيات أعلاه نتبع الخطوات التالية:

1. لا يتم التصريح بأربع متغيرات بغرض تخزين المعطيات العددية أعلاه.
2. التصريح بـ 4 متغيرات (لأنه لدينا 4 عمليات) بغرض تخزين نتيجة عمليات حساب القيمة المطلقة. هذه المتغيرات هي $r1, r2, r3, r2$ موزعة كما يلي:
   - 2 متغيرات من النوع الصحيح لتخزين القيمة المطلقة للمعطيات الصحيحة 367 , 450–

Var r1 , r2 : integer ;

- 2 متغيرات من النوع الحقيقي لتخزين القيمة المطلقة للمعطيات الحقيقية 782.09– , 10.32

Var r3 , r4 : real ;

3. لا يتم تخزين المعطيات في المتغيرات.

4. القيام بعمليات حساب القيمة المطلقة و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r1 := ABS (–450 ) ;
r2 := ABS ( 367 ) ;
r3 := ABS ( 10.32 ) ;
r4 := ABS(–782.09 ) ;
```

5. نشر نتائج عمليات القيمة المطلقة المخزنة في المتغيرات $r1, r2, r3, r2$. يتم ذلك كما يلي:

```
Writeln ( r1 ) ;
Writeln ( r2 ) ;
Writeln ( r3 ) ;
Writeln ( r4 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program VALEUR_ABSOLUT ;
Var  r1 , r2 : integer ;
     r3 , r4  : real ;
begin
     r1 := ABS (-450 ) ;
     r2 := ABS ( 367 ) ;
     r3 := ABS ( 10.32 ) ;
     r4 := ABS(-782.09 ) ;
     Writeln ( r1 ) ;
     Writeln ( r2 ) ;
     Writeln ( r3 ) ;
     Writeln ( r4 ) ;
     readln;
end.
```

04.04.05 – اللوغريتم النبيري: الدالة LN( )

تقوم الدالة LN( ) بحساب **اللوغريتم النبيري** لمعطية عددية سواء كانت صحيحة أو حقيقية. يتم ذالك وفقا لطريقة الكتابة التالية:

Variable Réelle **:= LN(** Variable Numérique **) ;**

حيث أن:

- Variable Numérique: تمثل المتغيرة العددية التي نريد أن نحسب اللوغريتم النبيري لمحتواها. هذا المحتوى قد يكون من النوع الصحيح و قد يكون من النوع الحقيقي.
- **LN( ):** تمثل الدالة التي تقوم بحساب **اللوغريتم النبيري** لمحتوى المتغيرة العددية Variable Numérique.
- Variable Réelle: تمثل متغيرة عددية يتم فيها تخزين نتيجة عملية حساب **اللوغريتم النبيري**. المتغيرة Variable Réelle يجب أن تكون من النوع الحقيقي و هذا مهما كان نوع المتغيرة العددية Variable Numérique حقيقيا أو صحيحا.

بغرض توضيح كيفية حساب **اللوغريتم النبيري** بواسطة الدالة LN( ) نأخذ المثال التالي:

مثـــال 24.05:

يتم حساب مربع المعطيات العددية $x=12.55$ و $y=30$ و $z=100$ و $w=10.34$ كما يلي:

- $LN(x)$
- $LN(y)$
- $LN(z)$
- $LN(w)$

مثـــال 25.05:

لتكن لدينا المعطيات العددية الحقيقية و الصحيحة التالية:

124 , 13 , 23.105 , 34.56

بغرض كتابة البرنامج الذي يسمح بحساب اللوغريتم النبيري للمعطيات أعلاه يجب:

1. التصريح بـ 4 متغيرات (لأنه لدينا 4 معطيات) بغرض تخزين فيها المعطيات أعلاه، هذه المتغيرات موزعة كما يلي:

- 2 متغيرات حقيقية و لتكن $v1,v2$ لتخزين المعطيات الحقيقية 23.105 , 34.56. يتم ذلك كما يلي:

Var v1 , v2 : integer ;

- 2 متغيرات حقيقية و لتكن $v3,v4$ لتخزين المعطيات الحقيقية 124 , 13. يتم ذلك كما يلي:

Var v3 , v4 : real ;

2. التصريح بـ 4 متغيرات (لأنه لدينا 4 عمليات) حقيقية بغرض تخزين نتيجة عمليات حساب اللوغريتم النبيري. هذه المتغيرات هي $r1, r2, r3, r2$

- 2 متغيرات صحيحة و لتكن $r1, r2$ لتخزين مربع القيم الصحيحة 1000 , 45. يتم ذلك كما يلي:

```
Var  r1 , r2 , r3 , r4  : real ;
```

3. تخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

```
v1 := 34.56 ;
v2 := 23.105 ;
v3 := 13 ;
v4 := 124 ;
```

4. القيام بعمليات حساب اللوغريتم و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r1 := LN( v1 ) ;
r2 := LN( v2 ) ;
r3 := LN( v3 ) ;
r4 := LN( v4 ) ;
```

5. نشر نتائج عمليات حساب اللوغريتم النبيري و المخزنة في المتغيرات $r1, r2, r3, r2$. يتم ذلك كما يلي:

```
Writeln ( r1 ) ;
Writeln ( r2 ) ;
Writeln ( r3 ) ;
Writeln ( r4 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program LOGARITHME ;
Var  v1 , v2 , r1 , r2 , r3 , r4 : real ;
   v3 , v4  : integer ;
begin
   v1 := 34.56 ;
   v2 := 23.105 ;
   v3 := 13 ;
   v4 := 124 ;
```

```
    r1 := LN( v1 ) ;
    r2 := LN( v2 ) ;
    r3 := LN( v3 ) ;
    r4 := LN( v4 ) ;
    Writeln ( r1 ) ;
    Writeln ( r2 ) ;
    Writeln ( r3 ) ;
    Writeln ( r4 ) ;
    readln;
end.
```

ملاحظـة 10.05 :

مثلما يمكن للدالة **LN( )** حساب مربع **محتوى** متغيرة من النوع العددي، يمكن لها كذلك حساب مربع **معطيـة** من النوع العددي دون اللجوء إلى إدخال هذه المعطية في متغيرة ثم نقوم بعد ذلك حساب مربع هذه المتغيرة. يتم ذالك حسب طريقة الكتابة التالية:

Variable Réelle **:= LN(** Donnée Numérique **) ;**

حيث أن:

- Donnée Numérique: تمثل المعطية العددية التي نريد أن نحسب مربعها. هذه المعطية قد تكون من النوع الصحيح و قد تكون من النوع الحقيقي.
- Variable Réelle: تمثل المتغيرة العددية من النوع الحقيقي التي يتم فيها تخزين نتيجة عملية حساب اللوغريتم النبيري. هذه المتغيرة يشترط أن تكون من النوع الحقيقي و هذا مهما كان نوع محتوى المتغيرة Donnée Numérique سواء كان من النوع الصحيح أو من النوع الحقيقي.

مثـــال 26.05:

يتم حساب مربع القيم 3.45 و 12 و 1.5 كما يلي:

- $LN(3.45)$
- $LN(12)$
- $LN(1.5)$

مثـــال 27.05:

لتكن لدينا المعطيات العددية التالية:

15 , 900 , 100.302 , 7820.009

بغرض كتابة البرنامج الذي يسمح بحساب مربع المعطيات أعلاه نتبع الخطوات التالية:

1. لا يتم التصريح بأربع متغيرات بغرض تخزين المعطيات العددية أعلاه.
2. التصريح بـ 4 متغيرات (لأنه لدينا 4 عمليات) من عددية من النوع الحقيقي بغرض تخزين نتيجة عمليات حساب اللوغريتم النبيري. هذه المتغيرات هي $r1, r2, r3, r2$ موزعة كما يلي:
3. لا يتم تخزين المعطيات في المتغيرات.
4. القيام بعمليات حساب اللوغريتم و تخزين النتائج في المتغيرات الموافقة لكل نتيجة. يتم ذلك كما يلي:

```
r1 := LN( 15 ) ;
r2 := LN( 900 ) ;
r3 := LN( 100.302 ) ;
r4 := LN( 7820.009 ) ;
```

5. نشر نتائج عمليات اللوغريتم و المخزنة في المتغيرات $r1, r2, r3, r2$. يتم ذلك كما يلي:

```
Writeln ( r1 ) ;
Writeln ( r2 ) ;
Writeln ( r3 ) ;
Writeln ( r4 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program LOGARITHME ;
Var   r1 , r2 , r3 , r4 : real ;
begin
    r1 := LN( 15 ) ;
    r2 := LN( 900 ) ;
    r3 := LN( 100.302 ) ;
    r4 := LN( 7820.009 ) ;
    Writeln ( r1 ) ;
    Writeln ( r2 ) ;
    Writeln ( r3 ) ;
    Writeln ( r4 ) ;
    readln;

end.
```

05.04.05 – الجزء الصحيـح: الدالـة INT( )

تقوم الدالة **INT( )** بتقديم **الجزء الصحيح** لمعطية عددية من النوع الحقيقي. يتم ذالك وفقا لطريقة الكتابة التالية:

Variable Numérique Réelle **:= INT(** Variable Numérique **) ;**

حيث أن:

- Variable Numérique: تمثل المتغيرة العددية التي نريد أن نجد أو نحسب **الجزء الصحيح** لمحتواها. هذه المتغيرة قد تكون من النوع الصحيح و قد تكون من النوع الحقيقي.
- **INT( )** : تمثل الدالة التي تقوم بتقديم **الجزء الصحيح** لمحتوى المتغيرة العددية Variable Réelle.
- Variable Numérique Réelle: تمثل متغيرة عددية من النوع الحقيقي يتم فيها تخزين نتيجة عملية حساب **الجزء الصحيح**. هذه المتغيرة يجب أن تكون من النوع الحقيقي.

بغرض توضيح كيفية حساب **الجزء الصحيح** بواسطة الدالة INT( ) نأخذ المثال التالي:

مثـــال 28.05:

لتكن المعطيات العددية من النوع الحقيقي التالية:

123.345 , - 456.12 , - 567.8999 , 0.008

بغرض كتابة البرنامج الذي يسمح بحساب و إيجاد الجزء الصحيح للقيم الحقيقية أعلاه نتبع الخطوات التالية:

1. التصريح بـ 4 متغيرات من النوع الحقيقي و لتكن val_1,val_2,val_3,val_4 بغرض تخزين المعطيات أعلاه يتم ذلك كما يلي:

Var val_1,val_2,val_3,val_4 : real ;

2. التصريح بـ 4 متغيرات من النوع الحقيقي و لتكن val_11,val_22,val_33,val_44 و التي يتم فيها تخزين نتيجة حساب الجزء الصحيح يتم ذلك كما يلي:

Var val_11,val_22,val_33,val_44 : real ;

3. القيام بتخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى

val_1 := 123.345 ;
val_2 := -456.12 ;
val_3 := -567.8999 ;
val_4 := 0.008 ;

4. حساب الجزء الصحيح بواسطة الدالة ( )INT و تخزين النتائج في المتغيرات val_11,val_22,val_33,val_44

```
val_11 := INT (val_1) ;
val_22 := INT (val_2) ;
val_33 := INT (val_3) ;
val_44 := INT (val_4) ;
```

5. نشر نتائج عملية حساب الجزء الصحيح و المخزنة في المتغيرات val_11,val_22,val_33,val_44

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program PARTIE_ENTIERE ;
Var   val_1,val_2,val_3,val_4 : real ;
    val_11,val_22,val_33,val_44 : real ;
begin
    val_1 := 123.345 ;
    val_2 := -456.12 ;
    val_3 := -567.8999 ;
    val_4 := 0.008 ;
    val_11 := INT (val_1) ;
    val_22 := INT (val_2) ;
    val_33 := INT (val_3) ;
    val_44 := INT (val_4) ;
    Writeln (val_11) ;
    Writeln (val_22) ;
    Writeln (val_33) ;
    Writeln (val_44) ;
    readln;
end.
```

ملاحظـة 11.05 :

مثلما يمكن للدالة **( )INT** حساب **الجزء الصحيح لمحتوى** متغيرة من النوع الصحيح أو الحقيقي، يمكن لها كذلك حساب **الجزء الصحيح لمعطية** من النوع الصحيح أو الحقيقي دون اللجوء إلى إدخال هذه المعطية في متغيرة من النوع الحقيقي ثم نقوم بعد ذلك بحساب **الجزء الصحيح** لهذه المتغيرة. يتم ذالك حسب طريقة الكتابة التالية:

Variable Réelle **:= INT(** Donnée Numérique **) ;**

حيث أن:

- Donnée Numérique: تمثل المعطية العددية من النوع الصحيح أو الحقيقي التي نريد أن نحسب الجزء الصحيح. هذه المعطية قد تكون من النوع الصحيح أو من النوع الحقيقي .
- Variable Réelle: تمثل المتغيرة العددية من النوع الحقيقي التي يتم فيها تخزين نتيجة عملية حساب الجزء الصحيح. هذه المتغيرة يشترط أن تكون من النوع الحقيقي.

مثـــال 29.05:

يتم حساب الجزء الصحيح للقيم 103.457− و 120.456 و 1607.5432− و 500.908 كما يلي:

- $INT(-103.457)$
- $INT(120.456)$
- $INT(-1607.5432)$
- $INT(500.908)$

مثـــال 30.05:

لتكن لدينا المعطيات العددية التالية:

15 , 900.456 , 100.302 , -7820.009

بغرض كتابة البرنامج الذي يسمح بحساب و إيجاد الجزء الصحيح للقيم الحقيقية أعلاه نتبع الخطوات التالية:

1. التصريح بـ 4 متغيرات من النوع الحقيقي و لتكن val_11,val_22,val_33,val_44 و التي يتم فيها تخزين نتيجة حساب الجزء الصحيح. يتم ذلك كما يلي:

Var val_11,val_22,val_33,val_44 : real ;

2. حساب الجزء الصحيح بواسطة الدالة ( )INT و تخزين النتائج في المتغيرات val_11,val_22,val_33,val_44

val_11 := INT (15) ;
*val_22 := INT (*900.456*) ;*
*val_33 := INT (*100.302*) ;*
val_44 := INT (-7820.009*) ;*

3. نشر نتائج عملية حساب الجزء الصحيح و المخزنة في المتغيرات val_11,val_22,val_33,val_44

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program PARTIE_ENTIERE ;
Var  val_11,val_22,val_33,val_44 : real ;
begin
    val_11 := INT (15) ;
    val_22 := INT (900.456) ;
    val_33 := INT (100.302) ;
    val_44 := INT (-7820.009) ;
    Writeln (val_11) ;
    Writeln (val_22) ;
    Writeln (val_33) ;
    Writeln (val_44) ;
    readln;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
15
900
100
-7820
_
```

06.04.05 – الجزء الصحيـح: الدالـة ( )TRUNC

إضافة إلى الدالة **( )INT** التي تقوم بتقديم الجزء الصحيح لمعطية من النوع الحقيقي، هناك الدالة **( )TRUNC** التي تقوم كذالك بتقديم **الجزء الصحيح** لمعطية عددية من النوع الحقيقي. يتم ذالك وفقا لطريقة الكتابة التالية:

Variable Numérique Entière **:= TRUNC(** Variable Réelle **) ;**

حيث أن:

- Variable Réelle: تمثل المتغيرة العددية التي نريد أن نجد أو نحسب **الجزء الصحيح** لمحتواها. هذا المحتوى قد يجب أن يكون من النوع الحقيقي.
- **TRUNC( )** : تمثل الدالة التي تقوم بتقديم **الجزء الصحيح** لمحتوى المتغيرة العددية Variable Réelle.
- Variable Numérique Entière: تمثل متغيرة عددية من النوع الصحيح يتم فيها تخزين نتيجة عملية حساب **الجزء الصحيح**. هذه المتغيرة يجب أن تكون من النوع الصحيح.

ملاحظـة 12.05 :

الفرق بين الدالة **INT( )** و الدالة **TRUNC( )** ما يلي:

- نتيجة الدالة **INT( )** عبارة عن معطية من النوع الحقيقي.
- نتيجة الدالة **TRUNC( )** عبارة عن معطية من النوع الصحيح.

07.04.05 – الجزء الصحيـح: الدالـة ROUND( )

تقوم الدالة **ROUND ( )** بتقديم **الجزء الصحيح** الأقرب لمعطية عددية من النوع الحقيقي. يتم ذالك وفقا لطريقة الكتابة التالية:

Variable Numérique Entière **:= ROUND (** Variable Réelle **) ;**

حيث أن:

- Variable Réelle: تمثل المتغيرة العددية التي نريد أن نجد أو نحسب **الجزء الصحيح الأقرب** لمحتواها. هذا المحتوى قد يجب أن يكون من النوع الحقيقي.
- **ROUND ( )** : تمثل الدالة التي تقوم بتقديم **الجزء الصحيح الأقرب** لمحتوى المتغيرة العددية Variable Réelle.
- Variable Numérique Entière: تمثل متغيرة عددية من النوع الصحيح يتم فيها تخزين نتيجة عملية حساب **الجزء الصحيح الأقرب**. هذه المتغيرة يجب أن تكون من النوع الصحيح.

08.04.05 – الجزء الحقيقي : الدالـة FRAC( )

تقوم الدالة **FRAC( )** بتقديم **الجزء العشري** لمعطية عددية من النوع الحقيقي. يتم ذالك وفقا لطريقة الكتابة التالية:

Variable Réelle **:= FRAC(** Variable Numérique **) ;**

حيث أن:

- Variable Numérique: تمثل المتغيرة العددية التي نريد أن نجد أو نحسب **الجزء العشري** لمحتواها. هذه المتغيرة قد تكون صحيحة أو حقيقية.
- **FRAC( )** : تمثل الدالة التي تقوم بتقديم **الجزء العشري** لمحتوى المتغيرة العددية Variable Numérique.
- Variable Réelle: تمثل متغيرة عددية من النوع الحقيقي يتم فيها تخزين نتيجة عملية حساب **الجزء العشري**. هذه المتغيرة يجب أن تكون من النوع الحقيقي.

بغرض توضيح كيفية حساب **الجزء العشري** بواسطة الدالة FRAC( ) نأخذ المثال التالي:

مثـــال 31.05:

لتكن لدينا المعطيات العددية التالية:

150.789 , 345.0075 , 1200.345 , 2345.0678 , 3456

بغرض كتابة البرنامج الذي يسمح بحساب و إيجاد القيم الصحيحة للقيم الحقيقية أعلاه نتبع الخطوات التالية:

1. التصريح بمتغيرة من النوع الصحيح و لتكن *e1* لتخزين المعطية الصحيحة 3456 أعلاه

Var e1 : integer ;

2. التصريح بـ 4 متغيرات من النوع الحقيقي و لتكن val_1,val_2,val_3,val_4 بغرض تخزين المعطيات الحقيقية أعلاه يتم ذلك كما يلي:

Var val_1,val_2,val_3,val_4 : real ;

3. التصريح بـ 4 متغيرات من النوع الحقيقي و لتكن val_11,val_22,val_33,val_44,val_55 و التي يتم فيها تخزين نتيجة حساب الجزء الصحيح يتم ذلك كما يلي:

Var val_11,val_22,val_33,val_44,val_55 : real ;

4. القيام بتخزين المعطيات أعلاه في المتغيرات التي تم التصريح بها في الخطوة الأولى

e1 := 3456 ;
val_1 := 150.789 ;
val_2 := 345.0075 ;
val_3 := 1200.345 ;
val_4 := 2345.0678 ;

5. حساب الجزء الحقيقي بواسطة الدالة ( )FRAC و تخزين النتائج في المتغيرات *val_11,val_22,val_33,val_44,val_55*

```
val_11 := FRAC (val_1) ;
val_22 := FRAC (val_2) ;
val_33 := FRAC (val_3) ;
val_44 := FRAC (val_4) ;
val_55 := FRAC (e1) ;
```

6. نشر نتائج عملية حساب الجزء الحقيقي و المخزنة في المتغيرات *val_11,val_22,val_33,val_44,val_55*

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program PARTIE_DECIMALE ;
Var   val_1,val_2,val_3,val_4 : real ;
      val_11,val_22,val_33,val_44,val_55 : real ;
      e1 : integer ;
begin
    val_1 := 150.789 ;
    val_2 := 345.0075 ;
    val_3 := 1200.345 ;
    val_4 := 2345.0678 ;
    e1 := 3456 ;
    val_11 := FRAC (val_1) ;
    val_22 := FRAC (val_2) ;
    val_33 := FRAC (val_3) ;
    val_44 := FRAC (val_4) ;
    val_55 := FRAC (e1) ;
    Writeln (val_11) ;
    Writeln (val_22) ;
    Writeln (val_33) ;
    Writeln (val_44) ;
    Writeln (val_55) ;
    readln;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
0.789
0.0075
0.345
0.0678
0.0000
_
```

ملاحظـة 13.05 :

مثلما يمكن للدالة **FRAC( )** حساب **الجزء الحقيقي لمحتوى** متغيرة من النوع الحقيقي، يمكن لها كذلك حساب **الجزء الحقيقي لمعطيـة** من النوع الحقيقي دون اللجوء إلى إدخال هذه المعطية في متغيرة من النوع الحقيقي ثم نقوم بعد ذلك بحساب **الجزء الحقيقي** لهذه المتغيرة. يتم ذالك حسب طريقة الكتابة التالية:

Variable Réelle **:= FRAC(** Donnée Réelle **) ;**

حيث أن:

- Donnée Réelle: تمثل المعطية العددية من النوع الحقيقي التي نريد أن نحسب الجزء الصحيح. هذه المعطية يشترط أن تكون من النوع الحقيقي.
- Variable Réelle: تمثل المتغيرة العددية من النوع الحقيقي التي يتم فيها تخزين نتيجة عملية حساب الجزء الصحيح. هذه المتغيرة يشترط أن تكون من النوع الحقيقي.

مثـــال 32.05:

يتم حساب الجزء الصحيح للقيم -103.457 و 120.456 و -1607.5432 و 500.908 كما يلي:

- $FRAC(-103.457)$
- $FRAC(120.456)$
- $FRAC(-1607.5432)$
- $FRAC(500.908)$

مثـــال 33.05:

لتكن لدينا المعطيات العددية التالية:

-7820.009 , 100.302 , 900.456 , 15

بغرض كتابة البرنامج الذي يسمح بحساب و إيجاد الجزء الحقيقي للقيم الحقيقية أعلاه نتبع الخطوات التالية:

1. التصريح بـ 4 متغيرات من النوع الحقيقي و لتكن val_11,val_22,val_33,val_44 و التي يتم فيها تخزين نتيجة حساب الجزء الصحيح. يتم ذلك كما يلي:

```
Var  val_11,val_22,val_33,val_44 : real ;
```

2. حساب الجزء الحقيقي بواسطة الدالة ( )FRAC و تخزين النتائج في المتغيرات val_11,val_22,val_33,val_44

```
val_11 := FRAC (15) ;
val_22 := FRAC (900.456) ;
val_33 := FRAC (100.302) ;
val_44 := FRAC (-7820.009) ;
```

3. نشر نتائج عملية حساب الجزء الصحيح و المخزنة في المتغيرات val_11,val_22,val_33,val_44

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program PARTIE_DECIMALE ;
Var  val_11,val_22,val_33,val_44 : real ;
begin
    val_11 := FRAC (15) ;
    val_22 := FRAC (900.456) ;
    val_33 := FRAC (100.302) ;
    val_44 := FRAC (-7820.009) ;
    Writeln (val_11) ;
    Writeln (val_22) ;
    Writeln (val_33) ;
    Writeln (val_44) ;
    readln;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
0
456
302
009
_
```

الجدول أدناه يلخص بعض الدوال التي يمكن أن تطبق على المعطيات الصحيحة و المغطيات الحقيقية

تسمية الدالة	رمز الدالة	نوع نتيجة الدالة
الجيب	Sin( )	حقيقي
التجب	Cos( )	حقيقي
التظل	Arctan( )	حقيقي
الأس	Exp( )	حقيقي

05.05 – قانـون الأولويـة (التـدرج) في تنفيـذ العلامـات الحسابيـة:

La Règle De Priorité Dans l'Exécution Des Opérateurs Arithmétiques:

يقوم الحاسوب بتنفيذ العمليات أو العلامات الحسابية تبعا للأولوية التالية:

1. عملية (علامة) الأس.
2. عملية (علامة) الضرب و القسمة و القسمة الصحيحة.
3. عملية (علامة) الجمع و الطرح.

بمعني أنه إذا كانت لدينا عبارة أو عملية حسابية تحتوي على العديد من العلامات الحسابية، فإن الحاسوب يبدأ **أولا** بإنجاز علامة الأس (الرفع لقوة) ثم ينتقل **ثانيا** إلى إنجاز علامتي الضرب و القسمة ثم ينتقل **ثالثا** إلى إنجاز علامتي الجمع و الطرح.

بغرض توضيح هذه الأولوية في إنجاز العلامات نأخذ المثال 34.05 أدناه:

مثـــال 34.05:

لتكن العبارات الحسابية التالية:

1. $Y = 150 + 45 - 10 + 50$

2. $Y = 150 - 45 + 10 - 20$

3. $Y = 50 + 20 + 10^3 - 60$

4. $T1 = 3500 * 100 / 50 * 6$

5. $T2 = 3500 / 100 * 50 / 5$

6. $T2 = 3500 / 100 * 50 / 5 + 9^2$

7. $Y = 1500 + 45 * 10 + 40$

8. $Z = 350 - 12 * 5 + 45$

9. $K = 100 + 45000 / 10 - 10$

10. $T = 3500 - 1200 / 5 + 30$

11. $X = 200 - 150 + 24 / 4 - 12 * 2 + 45 / 9 - 20 / 4 + 10 - 3 * 3$

12. $X = 200 - 150 + 24 / 4 - 12 * 2 + 45 / 9 - 20 / 4 + 10 - 3 * 3 + 100 ** 3 + 5 ** 2$

1. $Y = 150 + 45 - 10 + 50$:

العبارة الحسابية تحتوي فقط على عمليتي جمع و الطرح و التي لها **نفس الأولوية** لذالك يقوم الحاسوب بإنجاز هذه العلامات **من اليسار إلى اليمين** على التوالي:

- العملية $150 + 45$ و التي تساوي 195
- العملية $195 - 10$ و التي تساوي 185
- العملية $185 + 50$ و التي تساوي 235

2. $Y = 150 - 45 + 10 - 20$:

العبارة الحسابية تحتوي فقط على عمليتي جمع و الطرح و التي لها **نفس الأولوية** لذالك يقوم الحاسوب بإنجاز هذه العلامات **من اليسار إلى اليمين** على التوالي:

- العملية $150 - 45$ و التي تساوي 105
- العملية $105 + 10$ و التي تساوي 115
- العملية $115 - 20$ و التي تساوي 95

3. $Y = 50 + 20 + 10^3 - 60$

العبارة الحسابية تحتوي على عملية (علامة) الجمع و الطرح و الأس و بما أن **علامة الأس لها أولوية أكبر و أعلى من علامتي الجمع و الطرح**، فإن الحاسوب يقوم بإنجاز أولا علامة الأس ثم يقوم بعد ذلك بإنجاز علامتي الجمع و الطرح التي لها نفس الأولوية **من اليسار إلى اليمين** أي أن الحاسوب يقوم بإنجاز على التوالي:

- العملية 10^3 و التي تساوي 1000
- العملية $20+50$ و التي تساوي 70
- العملية $1000+70$ و التي تساوي 1070
- العملية $1070-60$ و التي تساوي 1010

4. $T1=3500*100/50*6$

العبارة الحسابية تحتوي فقط على عمليتي الضرب و القسمة و التي لها **نفس الأولوية** لذالك يقوم الحاسوب بإنجاز هذه العلامات **من اليسار إلى اليمين** على التوالي:

- العملية $100*3500$ و التي تساوي 350000
- العملية $350000/50$ و التي تساوي 7000
- العملية $6*7000$ و التي تساوي 42000

5. $T2=3500/100*50/5$

العبارة الحسابية تحتوي على عمليتي الضرب و القسمة و التي لها **نفس الأولوية** لذالك يقوم الحاسوب بإنجاز هذه العلامات **من اليسار إلى اليمين** على التوالي:

- العملية $3500/100$ و التي تساوي 35
- العملية $50*35$ و التي تساوي 1750
- العملية $1750/5$ و التي تساوي 350

6. $T2=3500/100*50/5+9^2$

العبارة الحسابية تحتوي على القسمة و الضرب و الجمع و الأس و بما أن **علامة الأس لها أولوية أكبر و أعلى من العلامات الأخرى**، فإن الحاسوب يقوم أولا بإنجاز علامة الأس ثم يقوم بعد ذلك بإنجاز علامات القسمة و الضرب التي لها نفس الأولوية **من اليسار إلى اليمين** ثم بعد ذلك يقوم بإنجاز عملية الجمع أي أن الحاسوب يقوم بإنجاز على التوالي:

- العملية 9^2 و التي تساوي 81
- العملية $3500/100$ و التي تساوي 35
- العملية $50*35$ و التي تساوي 1750
- العملية $1750/5$ و التي تساوي 350
- العملية $81+350$ و التي تساوي 431

7. $Y = 1500 + 45 * 10 + 40$

العبارة الحسابية تحتوي على عمليتي الجمع و الضرب و بما أن **علامة الضرب لها أولوية أكبر و أعلى من علامة الجمع،** فإن الحاسوب يقوم بإنجاز أولا علامة الضرب ثم يقوم بعد ذلك بإنجاز علامتي الجمع التي لها نفس الأولوية **من اليسار إلى اليمين** أي أن الحاسوب يقوم بإنجاز على التوالي:

- العملية $45 * 10$ و التي تساوي 450
- العملية $450 + 1500$ و التي تساوي 1950
- العملية $40 + 1950$ و التي تساوي 1990

8. $Z = 350 - 12 * 5 + 45$

العبارة الحسابية تحتوي على عملية الجمع و الطرح و الضرب و بما أن **علامة الضرب لها أولوية أكبر و أعلى من علامتي الجمع و الطرح،** فإن الحاسوب يقوم بإنجاز أولا علامة الضرب ثم يقوم بعد ذلك بإنجاز علامتي الجمع و الطرح التي لها نفس الأولوية **من اليسار إلى اليمين** أي أن الحاسوب يقوم بإنجاز على التوالي:

- العملية $12 * 5$ و التي تساوي 60
- العملية $350 - 60$ و التي تساوي 290
- العملية $45 + 290$ و التي تساوي 335

9. $K = 100 + 45000 / 10 - 10$

العبارة الحسابية تحتوي على عملية الجمع و القسمة و الطرح و بما أن **علامة القسمة لها أولوية أكبر و أعلى من علامتي الجمع و الطرح،** فإن الحاسوب يقوم بإنجاز أولا علامة القسمة ثم يقوم بعد ذلك بإنجاز علامتي الجمع و الطرح التي لها نفس الأولوية **من اليسار إلى اليمين** أي أن الحاسوب يقوم بإنجاز على التوالي:

- العملية $45000 / 10$ و التي تساوي 4500
- العملية $4500 + 100$ و التي تساوي 4600
- العملية $4600 - 10$ و التي تساوي 4590

10. $T = 3500 - 1200 / 5 + 30$

العبارة الحسابية تحتوي على عملية الطرح و القسمة و الجمع و بما أن **علامة القسمة لها أولوية أكبر و أعلى من علامتي الجمع و الطرح**، فإن الحاسوب يقوم بإنجاز أولا علامة القسمة ثم يقوم بعد ذلك بإنجاز علامتي الطرح و الجمع التي لها نفس الأولوية **من اليسار إلى اليمين** أي أن الحاسوب يقوم بإنجاز على التوالي:

- العملية $1200/5$ و التي تساوي 240
- العملية $240-3500$ و التي تساوي 3260
- العملية $30+3260$ و التي تساوي 3290

11. $X = 200-150+24/4-12*2+45/9-20/4+10-3*3$

تتضمن العبارة الحسابية على:

- 3 علامات جمع +
- 3 علامات طرح −
- 2 علامة ضرب *
- 3 علامات قسمة /

عندما يقوم الحاسوب بحساب نتيجة العبارة الحسابية أعلاه و التي تمثل قيمة المتغيرة X فإنه يبدأ **أولا** بإنجاز علامات الضرب و القسمة لأن لها أولوية أكبر ثم ينتقل **ثانيا** إلى إنجاز علامات الجمع و الطرح كما يلي:

1. إنجاز علامات الضرب و القسمة:

علامات الضرب و القسمة لها نفس الأولوية فيما بينها و بالتالي سوف يتم إنجازها **من اليسار إلى اليمين** و عليه سوف يتم إنجاز على التوالي ما يلي:

- العملية $24/4$ و التي تساوي 6
- العملية $2*12$ و التي تساوي 24
- العملية $45/9$ و التي تساوي 5
- العملية $20/4$ و التي تساوي 5
- العملية $3*3$ و التي تساوي 9

$X = 200-150+24/4-12*2+45/9-20/4+10-3*3$

2. إنجاز علامات الجمع و الطرح:

علامات الجمع و الطرح لها نفس الأولوية فيما بينها و بالتالي سوف يتم إنجازها **من اليسار إلى اليمين** و عليه سوف يتم إنجاز على التوالي ما يلي:

- العملية $150-200$ و التي تساوي 50
- العملية $6+50$ و التي تساوي 56
- العملية $24-56$ و التي تساوي 32
- العملية $5+32$ و التي تساوي 37
- العملية $5-37$ و التي تساوي 32
- العملية $10+32$ و التي تساوي 42
- العملية $9-42$ و التي تساوي 33

12. $X = 200-150+24/4-12*2+45/9-20/4+10-3*3+100**3+5**2$

تتضمن العبارة الحسابية على:

- 3 علامات جمع +
- 3 علامات طرح −
- 2 علامة ضرب *
- 3 علامات قسمة /
- 2 علامة أس **

عندما يقوم الحاسوب بحساب نتيجة العبارة الحسابية أعلاه و التي تمثل قيمة المتغيرة X فإنه يبدأ **أولا** بإنجاز علامات الأس لأن لها أولوية أكبر ثم ينتقل **ثانيا** إلى إنجاز علامات الضرب و القسمة ثم ينتقل **ثالثا** إلى إنجاز علامات الجمع و الطرح كما يلي:

1. إنجاز علامات الأس:

يتم إنجازها **من اليسار إلى اليمين** و عليه سوف يتم إنجاز على التوالي ما يلي:

- العملية 3**100 و التي تساوي 1000000
- العملية 2**5 و التي تساوي 25

2. إنجاز علامات الضرب و القسمة:

علامات الضرب و القسمة لها نفس الأولوية فيما بينها و بالتالي سوف يتم إنجازها **من اليسار إلى اليمين** و عليه سوف يتم إنجاز على التوالي ما يلي:

- العملية $24/4$ و التي تساوي 6
- العملية $12*2$ و التي تساوي 24
- العملية $45/9$ و التي تساوي 5
- العملية $20/4$ و التي تساوي 5
- العملية $3*3$ و التي تساوي 9

$$X = 200-150+24/4-12*2+45/9-20/4+10-3*3$$

3. إنجاز علامات الجمع و الطرح:

علامات الجمع و الطرح لها نفس الأولوية فيما بينها و بالتالي سوف يتم إنجازها **من اليسار إلى اليمين** و عليه سوف يتم إنجاز على التوالي ما يلي:

- العملية $200-150$ و التي تساوي 50
- العملية $6+50$ و التي تساوي 56
- العملية $56-24$ و التي تساوي 32
- العملية $5+32$ و التي تساوي 37
- العملية $37-5$ و التي تساوي 32
- العملية $10+32$ و التي تساوي 42
- العملية $42-9$ و التي تساوي 33
- العملية $1000000+33$ و التي تساوي 1000033
- العملية $1000033+25$ و التي تساوي 1000078

مثـــال 35.05:

لتكن العبارة الحسابية التالية:

$$Z := 1500+5000/(45+5)$$

في هذه الحالة أي حالة وجود الأقواس، فإن الحاسوب يقوم أولا بإنجاز ما بداخل الأقواس ثم ثانيا القسمة ثم ثالثا الجمع أي يقوم على التوالي بإنجاز ما يلي:

- العملية $(45+5)$ و التي تساوي 50
- العملية $5000/50$ و التي تساوي 100

- العملية $100+1500$ و التي تساوي 1600

من خلال المثال 35.05 أعلاه يمكن تقديم قانون الأولوية التالي:

1. الأقواس
2. عملية (علامة) الأس
3. عملية (علامة) الضرب و القسمة و القسمة الصحيحة
4. عملية (علامة) الجمع و الطرح

06.05 – ترجمة العبارات الحسابية إلى لغة الـ PASCAL:

من خلال هذه الفقرة سوف بتم التطرق إلى كيفية كتابة مختلف العبارات الحسابية بلغة PASCAL

الجدول أدناه يقدم بعض العبارات الحسابية و كيفية كتابتها أو ترجمتها إلى لغة PASCAL

العمليـة الحسابيـة	الكتابـة بلغة PASCAL
$A = 4790 + 2389 - 1000$	$A := 4790 + 2389 - 1000$
$S = 1500 - 2300 + 4200$	$S := 1500 - 2300 + 4200$
$c = 678 \times 55 - 100 \times 2$	$c := 678 * 55 - 100 * 2$
$X = 500{,}45 + 1250^2 \times 55 - 2{,}3$	$X := 500.45 + sqr(1250) * 55 - 2.3$
$r = 5 - \sqrt{160} \times 2{,}09$	$r := 5 - sqrt(160) * 2.09$
$N = \frac{134}{90} + \frac{78}{3} + 90 \times 3$	$N := 134/90 + 78/3 + 90 * 3$
$M = \frac{12 + 567}{45} + 150 \times \frac{2}{3} - 100$	$M := (12 + 567)/45 + 150 * 2/3 - 100$
$O = \sqrt{\frac{90 + 40}{24}} + \left(\frac{2}{3}\right)^2 + 3400$	$O := sqrt((90 + 40)/24) + sqr(2/3) + 3400$
$P = \sqrt{\lvert -123 \rvert} + 7890$	$P := sqrt(abs(-123)) + 7890$
$Z = 1500 + \frac{5000}{45 + 5}$	$Z := 1500 + 5000/(45 + 5)$

07.05 – عمليـات العلاقـات و علامـات العلاقـات:

Les Opérations Arithmétiques et Les Opérateurs De Relations :

تتمثل عمليات العلاقات فيما يلي:

1. علاقة الأكبر >
2. علاقة الأصغر <
3. علاقة المساواة =
4. علاقة اللامساواة <>
5. علاقة الأكبر أو يساوي =<
6. علاقة الأصغر أو يساوي =>

08.05 – العمليـات المنطقيـة و العلامـات المنطقيـة:

Les Opérations Logiques et Les Opérateurs Logiques :

تتمثل عمليات العلاقات فيما يلي:

1. الربط and
2. النفي not
3. الفصل or

تمـاريـن الفصـــل

✍ التمــريــن 01.05:

1. ماذا نعني بمعالجة المعطيات Traitement Des Données.

✍ التمــريــن 02.05:

1. أذكر مختلف العمليات الحسابية (Opérations Arithmétiques).
2. ما هي العلامات أو العوامل (Les Opérateurs) الموافقة لكل عملية حسابية متوصل إليها في السؤال الأول.

✍ التمــريــن 03.05:

1. أذكر مختلف العمليات المنطقية (Les Opérations Logiques).
2. ما هي العلامات أو العوامل (Les Opérateurs) الموافقة لكل عملية منطقية متوصل إليها في السؤال الأول.

✍ التمــريــن 04.05:

1. أذكر مختلف عمليات العلاقات (Les Opérations De Relations).
2. ما هي العلامات أو العوامل (Les Opérateurs) الموافقة لكل عملية علاقة متوصل إليها في السؤال الأول.

✍ التمــريــن 05.05:

1. حدد أولويات تنفيذ العلامات أو العوامل الحسابية المتوصل إليها في التمرين 02.05 أعلاه.

✍ التمــريــن 06.05:

لتكن المتغيرات العددية التالية:

$$a = 100$$

$$b = 25$$

$$c = 40$$

و لتكن العمليات الحسابية التالية:

```
d := 100 + a ;
a := a + b ;
c := c + b + 20*a ;
a := 2*a + 5*c ;
```

```
c := a ;
b := 5*c + 10*a + b ;
```

- **المطلـوب:**

1. انطلاقا من كل من المعطيات و العمليات الحسابية أعلاه حدد القيم النهائية للمتغيرات التالية:

- a
- b
- c
- d
- e

✍ التمـريـن 07.05:

لتكن المتغيرات العددية التالية:

$$a = 45$$
$$b = 25$$
$$c = 12$$

و لتكن العمليات الحسابية التالية:

```
d:= 100 + a ;
a:= a + d ;
e:= d + a ;
c:= d + a ;
f:= c + f – 2*a ;
f:= a + 45*f ;
```

- **المطلـوب:**

2. انطلاقا من كل من المعطيات و العمليات الحسابية أعلاه حدد القيم النهائية للمتغيرات التالية:

- a
- b
- c
- d
- e

- f

✍ التمـريــن 08.05:

لتكن المتغيرات العددية التالية:

$$w = 0.5$$
$$x = 250$$
$$y = 150$$
$$z = 100$$

و لتكن العمليات الحسابية التالية:

r:= z + y*w ;
t:= r + 100/w ;
z:= z – x/w + y ;
z := x + z ;
x := x – 2*z + 5*x ;
y := y – 10*y – 8x ;

- **المطلـوب:**

1. انطلاقا من كل من المعطيات و العمليات الحسابية أعلاه حدد القيم النهائية للمتغيرات التالية:

- x
- y
- z
- w
- r

✍ التمـريــن 09.05:

لتكن العبارات الحسابية التالية:

1. $k = 150 + 45 - 10 + 50 \times 10 + 12^3 + 5^5$

2. $k = 150 + 45 - 5(10 + 50)^3 \times 10 + 12^3 + 5^5$

3. $k = 150 + 45 - 5\dfrac{(10+50)^3}{45} \times 10 + \dfrac{12^3}{3+7} + 5^5$

4. $Y3 = 5 \times 4 - \dfrac{50 + 20 + 10^3 - 60}{4} + 12$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. بالإعتماد على مبدأ الأولوية، بين بالتفصيل كيفية إنجاز الحاسوب لعلامات العبارات الحسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 10.05:

لتكن العبارات الحسابية التالية:

1. $Y1 = 150 + 45 - 10 + 50$
2. $Y2 = 150 - 45 + 10 - 20$
3. $Y3 = 50 + 20 + 10^3 - 60$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. بالإعتماد على مبدأ الأولوية، بين بالتفصيل كيفية إنجاز الحاسوب لعلامات العبارات الحسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 11.05:

لتكن العبارات الحسابية التالية:

1. $T1 = 3500 \times \frac{100}{50} \times 6$
2. $T2 = \frac{3500}{100} \times \frac{50}{5}$
3. $T3 = \frac{3500}{100} \times \frac{50}{5} + 9^2$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. بالإعتماد على مبدأ الأولوية، بين بالتفصيل كيفية إنجاز الحاسوب لعلامات العبارات الحسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 12.05:

لتكن العبارات الحسابية التالية:

1. $Y = 1500 + 45 * 10 + 40$
2. $Z = 350 - 12 \times 5 + 45$
3. $K = 100 + \frac{45000}{10} - 10$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. بالإعتماد على مبدأ الأولوية، بين بالتفصيل كيفية إنجاز الحاسوب لعلامات العبارات الحسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 13.05:

لتكن العبارات الحسابية التالية:

1. $T = 3500 - \frac{1200}{5} + 30$
2. $X = 200 - 150 + \frac{24}{4} - 12 \times 2 + \frac{45}{9} - \frac{20}{4} + 10 - 3 \times 3$
3. $X1 = 200 - 150 + \frac{24}{4} - 12 \times 2 + \frac{45}{9} - \frac{20}{4} + 10 - 3 \times 3 + 100^3 + 5^2$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. بالإعتماد على مبدأ الأولوية، بين بالتفصيل كيفية إنجاز الحاسوب لعلامات العبارات الحسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 14.05:

لتكن لدينا المتغيرات العددية التالية:

$a = 1500$

$b = 500$

$c = 10$

$d = 5$

$e = 4.5$

و لتكن العمليات الحسابية التالية:

1. $x_1 = a + b$
2. $x_2 = c + d + e$
3. $x_3 = a + b + c + d + e$
4. $x_4 = a - e$
5. $x_5 = a - b - c - e$
6. $x_6 = a - b - c - d - e$
7. $x_7 = a + b - c + d - e$
8. $x_8 = a - b + c - d + e$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أحسب نتيجة كل عملية حسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 15.05:

لتكن لدينا المعطيات العددية الحقيقية و الصحيحة التالية

205.15 , 875 , 200.34 , 105.45 , 556 , 400 , 500.67

حيث نريد إنجاز و إجراء عمليات الجمع التالية:

1. $875 + 400$ و تخزين نتيجة الجمع في المتغيرة $r26$.
2. $875 + 556 + 400$ و تخزين النتيجة في $r236$.
3. $205.15 + 200.34$ و تخزين النتيجة في $r57$.
4. $205.15 + 200.34 + 105.45 + 500.67$ و تخزين النتيجة في $r1457$.
5. $400 + 500.67$ و تخزين النتيجة في $r12$.
6. جمع جميع القيم و تخزين النتيجة في r.

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 16.05:

لتكن المعطيات الصحيحة و الحقيقية التالية

450.234 , 45.578 , 100.45 , 4500 , 1500 , 120

نريد إجراء عمليات الجمع التالية:

1. $120+1500+4500$
2. $100.45+45.578+450.234$
3. $120+1500+4500+450.234$
4. $120+1500+4500+100.45+45.578+450.234$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 17.05:

لتكن لدينا المعطيات العددية الصحيحة و الحقيقية التالية:

$$123\ ,\ 450.34\ ,\ 1250\ ,\ 900.45$$

نريد إنجاز و إجراء عمليات الطرح التالية:

1. $123-1250$ و تخزين نتيجة الطرح في المتغيرة $r31$.
2. $1250-123$ و تخزين النتيجة في المتغيرة $r13$.
3. $450.34-900.45$ و تخزين النتيجة في $r42$.
4. $900.45-1250$ و تخزين النتيجة في $r34$.
5. $900.45-1250-450.34-123$ و تخزين النتيجة في r.

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 18.05:

لتكن المعطيات الصحيحة و الحقيقية التالية

$$450.234\ ,\ 45.578\ ,\ 100.45\ ,\ 4500\ ,\ 1500\ ,\ 120$$

نريد إجراء عمليات الطرح التالية:

1. $120-1500-4500$
2. $100.45-45.578-450.234$
3. $120-1500-4500-450.234$
4. $120-1500-4500-100.45-45.578-450.234$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمــريــن 19.05:

لتكن المتغيرات العددية من النوع الصحيح و من النوع الحقيقي التالية:

$$k = 2.5$$
$$l = 200$$
$$m = 8$$
$$n = 4$$

و لتكن العمليات الحسابية التالية:

1. $v1 = m \times k$
2. $v2 = m \times k + l - 10k$
3. $v3 = 1.5 \times n \times m + 0.5 \times l$
4. $v4 = (k + m)0.5 + (l + k)k$
5. $v5 = (n + m - l \times k) + 10m - 2k$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أحسب نتيجة كل عملية حسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمــريــن 20.05:

لتكن لدينا المعطيات العددية الصحيحة و الحقيقية التالية:

15 , 3.75 , 15.05 , 4

نريد إنجاز و إجراء عمليات الضرب التالية:

1. 4×15 و تخزين نتيجة الضرب في المتغيرة $r14$.
2. 15×3.75 و تخزين النتيجة في المتغيرة $r12$.
3. 4×15.05 و تخزين النتيجة في المتغيرة $r43$.
4. 15.05×3.75 و تخزين النتيجة في المتغيرة $r23$.
5. $4 \times 15.05 \times 3.75 \times 15$ و تخزين النتيجة في المتغيرة r.
6. $5 \times 10 \times 4 \times 15.05 \times 3.75 \times 15$ و تخزين النتيجة في المتغيرة rr.

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمــريــن 21.05:

لتكن المعطيات الصحيحة و الحقيقية التالية

120 , 1500 , 4500 , 100.45 , 45.578 , 450.234

نريد إجراء عمليات الضرب التالية:

1. $120*1500*4500$
2. $100.45*45.578*450.234$
3. $120*1500*4500*450.234$
4. $120*1500*4500*100.45*45.578*450.234$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمــريــن 22.05:

لتكن المتغيرات العددية من النوع الصحيح و من النوع الحقيقي التالية:

$$r = 40$$
$$s = 10.5$$
$$t = 100$$
$$u = 50$$

و لتكن العمليات الحسابية التالية:

1. $w_1 = \frac{t}{2} + \frac{r}{10} - 15$
2. $w_2 = u + \frac{r}{s} - \frac{t}{2} + 10$
3. $w_3 = 1000 + \frac{u+t}{s} + 10$
4. $w_4 = 5 + \frac{r+s}{t+5} + \frac{10u}{s+u} - 50s$
5. $w_4 = 5s + \frac{r+s+5+10t}{r+u} + \frac{10u+2u}{s+u} - 5ts$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أحسب نتيجة كل عملية حسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمــريــن 23.05:

لتكن المتغيرات العددية من النوع الصحيح و من النوع الحقيقي التالية:

$$w = 1.5$$
$$x = 100$$
$$y = 150$$
$$z = 10$$

و لتكن العمليات الحسابية التالية:

1. $a_1 = x + w + z$
2. $a_2 = w - x + z$
3. $a_3 = z + 2x + w$
4. $a_4 = y - 5w + 12x$
5. $a_5 = 245x + 10z - 20x + z - 2y$
6. $a_6 = \frac{x}{z}$
7. $a_7 = \frac{y + x}{2}$
8. $a_8 = 5y + \frac{x}{2} + 2x - w$
9. $a_9 = 12w + \frac{20z}{x} - 5y$
10. $a_10 = \frac{w + 50.5}{2x}$
11. $a_11 = \frac{x + 120}{5y + 15}$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أحسب نتيجة كل عملية حسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمــريــن 24.05:

لتكن المتغيرات التالية:

$$x = 2$$
$$y = 4$$

$z = 8$

و لتكن العمليات الحسابية التالية:

1. $a = x + y + 1500$
2. $a_1 = 2400 - x + y$
3. $b = 2x + 123 - 4y$
4. $a2 = 1000x - 500x + 400y$
5. $e = \frac{x}{y}$
6. $m = \frac{x + y + z}{3}$
7. $n = \frac{5x + 8y + 2z}{x - 5y}$
8. $n1 = \frac{3400}{3z} + \frac{5x + 8y + 2z}{x - 5y}$
9. $r = \frac{120x}{15} + 34z + \frac{4y}{5}$
10. $r3 = x + z + 1200 - \frac{400y}{x}$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أحسب نتيجة كل عملية حسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 25.05:

لتكن لدينا المعطيات العددية الصحيحة و الحقيقية التالية:

150 , 4 , 500.50 , 44.88

نريد كتابة البرنامج الذي يسمح بإنجاز و إجراء عمليات القسمة التالية:

1. $4 \div 150$ و تخزين نتيجة القسمة في المتغيرة $r12$.
2. $150 \div 4$ و تخزين النتيجة في المتغيرة $r21$.
3. $4 \div 44.88$ و تخزين النتيجة في المتغيرة $r42$.
4. $44.88 \div 500.50$ و تخزين النتيجة في المتغيرة $r34$.
5. $4 \div 500.50$ و تخزين النتيجة في المتغيرة $r32$.

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 26.05:

لتكن العبارات الحسابية التالية:

1. $12 \ div \ 3$

2. $13 \ div \ 3$

3. $14 \ div \ 3$

4. $16 \ div \ 4$

5. $15 \ div \ 3$

6. $14 \ div \ -3$

7. $-14 \ div \ -3$

8. $16 \ div \ -4$

9. $-16 \ div \ -4$

10. $-200 \ div \ 25$

- **المطلـوب:**

1. حدد نتيجة كل عملية حسابية أعلاه

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 27.05:

لتكن العبارات الحسابية التالية:

1. $5 \ \text{mod} \ 6$

2. $-5 \ \text{mod} \ 6$

3. $-6 \ \text{mod} \ 4$

4. $-6 \ \text{mod} \ 5$

5. $21 \ \text{mod} \ 3$

6. $14 \ \text{mod} \ 3$

7. $14 \ \text{mod} \ -3$

8. $16 \bmod -4$
9. $-16 \bmod -4$
10. $-200 \bmod 25$

- **المطلـوب:**

1. حدد نتيجة كل عملية حسابية أعلاه
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 28.05:

لتكن العمليات التالية:

1. $220 \bmod 70$
2. $70 \bmod 220$
3. $220 \ div \ 70$
4. $70 \ div \ 220$
5. $400 \ div \ 1500$
6. $1500 \ div \ 400$
7. $400 \bmod 1500$
8. $1500 \bmod 400$

- **المطلـوب:**

1. أحسب نتيجة كل عملية حسابية أعلاه.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 29.05:

لتكن لدينا التصريحات التالية:

```
Var  x , y , z : real ;
     a , b , c : integer ;
```

و لتكن العمليات التالية:

1. $x + y$
2. $z + x + y$
3. $z + a$

4. $a+x+c$
5. $a+b$
6. $a+b+y$
7. $x*y$
8. $a*x+y$
9. $z*y$
10. $a*b$
11. a/b
12. z/b
13. a/x
14. x/y
15. $b \ div \ a$
16. $a \ div \ b$
17. $a \ \text{mod} \ b$
18. $b \ \text{mod} \ a$
19. $x \ \text{mod} \ y$
20. $y \ \text{mod} \ x$

- **المطلـوب:**

1. حدد نوع معطيات العمليات الحسابية أعلاه.

✍ التمـريـن 30.05:

لتكن لدينا التصريحات التالية:

```
Var  x , y , z : real ;
     a , b , c : integer ;
```

و لتكن العمليات التالية:

$x = 100.5$

$y = 2.5$

$z = 10.5$

$a = 100$

$b = 2$

$c = 10$

و لتكن العمليات الحسابية للتمرين 29.05 أعلاه

- **المطلـوب:**

1. أحسب نتيجة كل عملية حسابية.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 31.05:

لتكن لدينا التصريحات التالية:

```
Const  f =12.5 ;
       m = 100 ;
Var  x , y , z : real ;
     a , b , c : integer ;
```

حيث أن:

$a = 30$

$b = 40$

$y = -10.5$

و لتكن العبارات الحسابية التالية:

1. $c := a \mod b$
2. $c := (9900 - m) \ div \ a$
3. $c := a \mod y$
4. $x := f * y$
5. $c := a / b$
6. $x := a / b$
7. $y := a \mod (a / b)$
8. $c := b \ div \ 0$
9. $c := a \mod (9900 - m)$
10. $c := a \mod 0$
11. $x := a \ div \ b$
12. $x := a * 2.5 / y$

13. $x := a*/b*a$
14. $x := (a+b)/(y+0.6)$

- **المطلـوب:**

1. حدد العبارات الحسابية الخاطئة من بين العبارات الحسابية أعلاه.
2. حدد نوع معطيات العبارات الحسابية الصحيحة المتوصل إليها.
3. أحسب نتيجة كل عبارة حسابية.

✍ التمـريـن 32.05:

لتكن العمليات الحسابية التالية:

1. $A = 4790 + 2389 - 1000$
2. $S = 1500 - 2300 + 4200$
3. $c = 678 \times 55 - 100 \times 2$
4. $X = 500{,}45 + 1250^2 \times 55 - 2{,}3$
5. $r = 5 - \sqrt{160} \times 2{,}09$
6. $N = \frac{134}{90} + \frac{78}{3} + 90 \times 3$
7. $M = \frac{12 + 567}{45} + 150 \times \frac{2}{3} - 100$
8. $O = \sqrt{\frac{90 + 40}{24}} + \left(\frac{2}{3}\right)^2 + 3400$
9. $Z = 1500 + \frac{5000}{45 + 5}$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أحسب نتيجة كل عملية حسابية أعلاه.
3. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 33.05:

لتكن لدينا المعطيات العددية الصحيحة و الحقيقية التالية:

5 , 0.5 , 5.4 , 3

نريد كتابة البرنامج الذي يسمح بإنجاز و إجراء عمليات القسمة التالية:

1. $5^{0.5}$ و تخزين نتيجة الضرب في المتغيرة $r12$.

2. $(0.5)^3$ و تخزين نتيجة الضرب في المتغيرة $r24$.
3. $(5.4)^5$ و تخزين نتيجة الضرب في المتغيرة $r31$.
4. $(0.5)^{5.4}$ و تخزين نتيجة الضرب في المتغيرة $r23$.
5. $(0.5+5.4)^5$ و تخزين نتيجة الضرب في المتغيرة rr.
6. $(5+0.5+3)^{5.4}$ و تخزين نتيجة الضرب في المتغيرة r.

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 34.05:

1. أذكر مختلف الدوال التي تطبق على المعطيات العددية.
2. قدم نوع المعطية التي تمثل دليل كل دالة متوصل إليها في المطلوب الأول.
3. قدم نوع المعطية التي تمثل نتيجة كل دالة متوصل إليها في المطلوب الأول.

✍ التمـريـن 35.05:

1. قدم الدالة التي تقوم بحساب الجذر التربيعي.
2. قدم طريقة الكتابة للدالة المتوصل إليها في المطلوب الأول.

✍ التمـريـن 36.05:

لتكن المعطيات العددية التالية:

$$15625\ ,\ 25\ ,\ 20.25\ ,\ 2550.25$$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب الجذر التربيعي للمعطيات أعلاه.

✍ التمـريـن 37.05:

لتكن المتغيرات العددية التالية:

$$a = 25$$
$$b = 49$$
$$c = 625$$
$$d = 100$$

نود إنجاز العمليات الحسابية التالية:

1. $\sqrt{a}$

2. $\sqrt{b}$
3. $\sqrt{c}$
4. $\sqrt{d}$
5. $\sqrt{\frac{a}{4}}$
6. $\sqrt{a+b}$
7. $\sqrt{a\times d}$
8. $\sqrt{a+b+c+d}$
9. $\sqrt{\frac{a\times d}{9}}$
10. $\sqrt{\frac{a+b+c+d}{a}}\times\sqrt{\frac{c}{a}+81}$
11. $\sqrt{\frac{a+b}{c+d}}$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمــريــن 38.05:

3. قدم الدالة التي تقوم بحساب المربع.
4. قدم طريقة الكتابة للدالة المتوصل إليها في المطلوب الأول.

✍ التمــريــن 39.05:

لتكن المعطيات العددية التالية:

45 , 1000 , 230.34 , 25.25

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب مربع المعطيات أعلاه.

✍ التمــريــن 40.05:

لتكن المتغيرات العددية التالية:

$a = 4$

$b = 5$

$c = 10$

$d = 9$

نود إنجاز العمليات الحسابية التالية:

1. a^2
2. b^2
3. c^2
4. d^2
5. $(a+3)^2$
6. $(b+c)^2$
7. $(a+c+d)^2$
8. $\left(\frac{c}{2}\right)^2$
9. $\left(\frac{a+d}{b+2}\right)^2$
10. $(a+b)^2+(a+c)^2+(5d)^2$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 41.05:

1. قدم الدالة التي تقوم بحساب القيمة المطلقة.
2. قدم طريقة الكتابة للدالة المتوصل إليها في المطلوب الأول.

✍ التمـريـن 42.05:

لتكن المعطيات العددية التالية:

$1250 \ , \ -4250 \ , \ 56.09 \ , \ -423.567$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب القيمة المطلقة أعلاه.

✍ التمـريـن 43.05:

لتكن المتغيرات العددية التالية:

$a = -10$

$b = -50$

$c = -10.9$

$d = 9.5$

نود إنجاز العمليات الحسابية التالية:

1. $|a|$
2. $|b|$
3. $|c|$
4. $|d|$
5. $|a-b|$
6. $|a-b-c|$
7. $|5b+2c|$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 44.05:

1. قدم الدالة التي تقوم بحساب اللوغريتم النبيري.
2. قدم طريقة الكتابة للدالة المتوصل إليها في المطلوب الأول.

✍ التمـريـن 45.05:

لتكن المعطيات العددية التالية:

124 , 13 , 23.105 , 34.56

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب اللوغريتم النبيري للمعطيات أعلاه.

✍ التمـريـن 46.05:

لتكن المتغيرات العددية التالية:

$x = 12.55$

$y = 30$

$z = 100$

$w = 10.34$

نود إنجاز العمليات الحسابية التالية:

1. $\ln\ x$
2. $\ln\ y$
3. $\ln\ z$
4. $\ln\ w$
5. $\ln\ (x+y)$
6. $\ln\ (z+w)$
7. $\ln\ (x+1)$
8. $\ln\ (x+y+5)$
9. $\ln\ (2x+3y)$
10. $\ln\left(\frac{x+3y}{2}+5\right)$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 47.05:

1. قدم الدالة التي تقوم بتقديم الجزء الصحيح.
2. قدم طريقة الكتابة للدالة المتوصل إليها في المطلوب الأول.

✍ التمـريـن 48.05:

لتكن المعطيات العددية التالية:

124 , 13 , 23.105 , 34.56

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الجزء الصحيح للمعطيات أعلاه.

✍ التمـريـن 49.05:

لتكن المتغيرات العددية التالية:

$$x = -512.155$$
$$y = 30.478$$
$$z = 100$$
$$w = -10.34$$

نود إيجاد الجزء الصحيح لما يلي:

1. x
2. y
3. z
4. w
5. $(x+y)$
6. $(z+w)$
7. $(x+1)$
8. $(x+y+5)$
9. $(2x+3y)$
10. $\left(\frac{x+3y}{2}+5\right)$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح إيجاد الجزء الصحيح للعبارات أعلاه.

✍ التمـريـن 50.05:

1. قدم الدالة التي تقوم بتقديم الجزء الحقيقي.
2. قدم طريقة الكتابة للدالة المتوصل إليها في المطلوب الأول.

✍ التمـريـن 51.05:

لتكن المعطيات العددية التالية:

$$345\ ,\ -123.567\ ,\ 556.0097\ ,\ -42.50\ ,\ 1250.345\ ,\ 4567$$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الجزء الحقيقي للمعطيات أعلاه.

✍ التمـريــن 52.05:

لتكن المتغيرات العددية التالية:

$$x = 4500.456$$
$$y = -223.9007$$
$$z = 100.78602$$
$$w = 500$$
$$l = 1240$$

نود إيجاد الجزء الحقيقي لما يلي:

1. x
2. y
3. z
4. w
5. l
6. $x + y$
7. $x + y + l$
8. $2x + 5y$
9. $w + l$
10. $5w + 6l$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح إيجاد الجزء الحقيقي للعبارات أعلاه.

✍ التمـريــن 53.05:

لتكن العبارات التالية:

1. $a = (45x + 12y)^5$
2. $s = \dfrac{(45x + 12y)^5}{(z + 400)}$
3. $a = \sqrt{|2x + yx|}$
4. $b = \sqrt{230 + \left|\dfrac{5y}{4x + 3y}\right|}$

5. $a_1 = 125x + \frac{x(y-z)^2}{5(6y+12x)^4}$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 54.05:

لتكن المتغيرات التالية:

$$w = 2$$
$$x = 5$$
$$y = 10$$
$$z = 4$$

و لتكن العمليات الحسابية التالية:

1. $a_1 = x + z + w + y$
2. $a_2 = 5x + 6z + 10w + 12y$
3. $a_3 = 10x^2 + 23z^4 + w - y$
4. $a_4 = \frac{x+z+w+y}{4} + 450$
5. $a_5 = (x+z+w+y)^2$
6. $a_6 = \sqrt{45+2x} + |x+z+w+y|$
7. $a_7 = \frac{2x+3z}{45+y} + (w+y)^3$
8. $a_8 = \left(\frac{x+z}{z-w}\right)^4 + \frac{10w^5}{5} + 45y^3$
9. $a_9 = \sqrt{\frac{x+z+w+y}{4+4x+7y}}$
10. $a_10 = \left|\frac{200x+30z+90w+40y^4}{\sqrt{w+z}}\right|$

- **المطلـوب:**

1. أكتب العبارات الحسابية أعلاه بلغة الـ PASCAL.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإنجاز العمليات الحسابية أعلاه.

✍ التمـريـن 55.05:

لتكن المتغيرات التالية:

$$x1 = 30$$
$$x2 = 20$$
$$x3 = 10$$
$$x4 = 20$$
$$x5 = 50$$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب ما يلي:

- مجموع القيم: $som1 = x1 + x2 + x3 + x4 + x5$
- مربع مجموع القيم: $som2 = (x1 + x2 + x3 + x4 + x5)^2$
- مجموع مربعات القيم: $som3 = x1^2 + x2^2 + x3^2 + x4^2 + x5^2$
- مجموع مكعبات القيم: $som4 = x1^3 + x2^3 + x3^3 + x4^3 + x5^3$

✍ التمـريـن 56.05:

لتكن المتغيرات التالية:

$$x1 = 300$$
$$x2 = 200$$
$$x3 = 100$$
$$x4 = 200$$
$$x5 = 500$$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب ما يلي:

- المتوسط الحسابي moy_Ar La Moyenne Arithmétique حيث أن:

$$moy_ar = \frac{x1 + x2 + x3 + x4 + x5}{5}$$

- حساب المتوسط الهندسي moy_Ge La Moyenne Géométrique حيث أن:

$$moy_Ge = \sqrt[5]{Y_1 \times_2 \times Y_3 \times Y_4 \times Y_5}$$

- حساب الانحراف المتوسط ec_moy l'Ecart Moyen حيث أن:

$$ec_moy = \frac{|x1 - moy_ar| + |x2 - moy_ar| + ... + |x5 - moy_ar|}{5}$$

- حساب التباين $Varx$ La Variance حيث أن:

$$\text{var } x = \frac{(x1 - moy_ar)^2 + (x2 - moy_ar)^2 + \ldots\ldots\ldots\ldots + (x5 - moy_ar)^2}{5}$$

التمـريـن 57.05:

لتكن المتغيرات التالية:

$$y1 = 30$$
$$y2 = 20$$
$$y3 = 10$$
$$y4 = 20$$
$$y5 = 50$$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب ما يلي:

- حساب العزم البسيط من الدرجة الأولى M_S^1 للمتغير Y حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 1\ : M_S^1 = \frac{Y_1 + Y_2 + Y_3 + Y_4 + Y_5}{5}$$

- حساب العزم البسيط من الدرجة الثانية M_S^2 للمتغير Y حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 2\ : M_S^2 = \frac{Y_1^2 + Y_2^2 + Y_3^2 + Y_4^2 + Y_5^2}{5}$$

- حساب العزم البسيط من الدرجة الثالثة M_S^3 للمتغير Y حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 3\ : M_S^3 = \frac{Y_1^3 + Y_2^3 + Y_3^3 + Y_4^3 + Y_5^3}{5}$$

- حساب العزم البسيط من الدرجة الرابعة M_S^4 للمتغير Y حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 4\ : M_S^4 = \frac{Y_1^4 + Y_2^4 + Y_3^4 + Y_4^4 + Y_5^4}{5}$$

التمـريـن 58.05:

الجدول أدناه يقدم خمسة قيم للمتغير Y

قيم المتغير Y
$Y_1 = 5$
$Y_2 = 2$
$Y_3 = 1$
$Y_4 = 2$
$Y_5 = 3$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب ما يلي:

- المتوسط الحسابي $\overline{Y}$ La Moyenne Arithmétique حيث أن:

$$\overline{Y} = \frac{Y_1 + Y_2 + Y_3 + Y_4 + Y_5}{5}$$

- حساب العزم المركزي من الدرجة الأولى M_C^1 للمتغير Y حيث أن:

$$M_C^1 = \frac{\left(Y_1 - \overline{Y}\right)^1 + \left(Y_2 - \overline{Y}\right)^1 + \left(Y_3 - \overline{Y}\right)^1 + \left(Y_4 - \overline{Y}\right)^1 + \left(Y_5 - \overline{Y}\right)^1}{5}$$

- حساب العزم المركزي من الدرجة الثانية M_C^2 للمتغير Y حيث أن:

$$M_C^2 = \frac{\left(Y_1 - \overline{Y}\right)^2 + \left(Y_2 - \overline{Y}\right)^2 + \left(Y_3 - \overline{Y}\right)^2 + \left(Y_4 - \overline{Y}\right)^2 + \left(Y_5 - \overline{Y}\right)^2}{5}$$

- حساب العزم المركزي من الدرجة الثالثة M_C^3 للمتغير Y حيث أن:

$$M_C^3 = \frac{\left(Y_1 - \overline{Y}\right)^3 + \left(Y_2 - \overline{Y}\right)^3 + \left(Y_3 - \overline{Y}\right)^3 + \left(Y_4 - \overline{Y}\right)^3 + \left(Y_5 - \overline{Y}\right)^3}{5}$$

الفصــل 📖 الســادس

معـالجــة المعطيــات سلسلــة أحــرف

Traitement Des Données Chaine De Caractères

أهـداف الفصـل:

بعد انتهائك من دراسـة و الإطلاع بعنـاية على محتويات هذا الفصل، فإنك تستطيـع الإلمـام بمـا يلـي:

- مختلف العمليات التي تطبق على المعطيات من النوع سلسلة أحرف.
  1. تغيير معطية من النوع سلسلة أحرف
  2. ربط المعطيات من النوع سلسلة أحرف
     - الربط باستخدام العلامة +
     - الربط باستخدام الدالة Concat( )
  3. استخلاص مجموعة جزئية الدالة COPY( )
  4. حذف مجموعة جزئية الدالة DELETE( )
  5. طول معطية سلسلة أحرف الدالة LENGTH( )
- مختلف الدوال التي تطبق على المعطيات سلسلة أحرف.
  1. الدالة INSERT( )
  2. الدالة Ord( )
  3. الدالة Chr( )
  4. الدالة Succ( )
  5. الدالة Pred( )

01.06 – تمهيـد:

مثلما يمكن معالجة المعطيات العددية، يمكن كذالك معالجة المعطيات من النوع سلسلة أحرف. من بين العمليات الممكن القيام بها على المعطيات من النوع سلسلة أحرف نذكر ما يلي:

1. تغيير المعطيات من النوع سلسلة أحرف Modification D'une Donnée Chaine De Caractères
2. ربط المعطيات من النوع سلسلة أحرف La Concaténation Des Données Chaine De Caractères
3. مقارنة المعطيات من النوع سلسلة أحرف La Comparaison Des Données Chaine De Caractères

02.06 – تغيير معطيـة من النـوع سلسلـة أحـرف:

Modification D'une Donnée Chaine De Caractères :

يمكن إجراء أي تغيير على أي معطية من النوع سلسلة أحرف، كأن نقوم باستبدال حرف من أحرفها بحرف آخر. بغرض التوضيح أكثر هذه التغيرات الممكن إجرائها على المعطيات من النوع سلسلة أحرف، نأخذ المثال التالي:

مثـــال 01.06:

لتكن لدينا المتغيرة من النوع سلسلة أحرف التالية: n, p, q, l و التي تساوي أو تحتوي على التوالي على المعطيات من النوع سلسلة أحرف التالية: $Z9HRA, SIHEM, AB + EL8ADE*, HAYETTI$ أي أن:

$$n = 'Z9HRA'$$
$$p = 'SIHEM'$$
$$q = 'AB + EL8ADE*'$$
$$l = 'HAYETTE'$$

لنفرض أننا نريد إجراء تغييرات على محتوى المتغيرات n, p, q, l أي إجراء تغييرات على المعطيات سلسلة أحرف $Z9HRA, SIHEM, AB + EL8ADE*, HAYETTE$ كما يلي:

1. التغييرات على المتغيرة $n = 'Z9HRA'$
   - استبدال الحرف رقم 2 الذي هو 9 بالحرف A و عليه يصبح محتوى المتغيرة n هو $ZAHRA$ بدلا من $Z9HRA$ أي أن $n = 'ZAHRA'$ بدلا من $n = 'Z9HRA'$
2. التغييرات على المتغيرة $p = 'SIHEM'$

- استبدال الحرف رقم 4 الذي هو E بالرمز * و عليه يصبح محتوى المتغيرة p هو $SIH*M$ بدلا من $SIHEM$ أي أن $p='SIH*M'$ بدلا من $p='SIHEM'$

3. التغييرات على المتغيرة $q='AB+EL8ADE*'$

- استبدال الحرف رقم 3 الذي هو + بالحرف D و عليه يصبح محتوى المتغيرة q هو $ABDEL8ADE*$ بدلا من $AB+EL8ADE*$ أي أن $q='ABDEL8ADE*'$ بدلا من $q='AB+EL8ADE*'$
- استبدال الحرف رقم 6 الذي هو 8 بالحرف K و عليه يصبح محتوى المتغيرة q هو $ABDELKADE*$ بدلا من $ABDEL8ADE*$ أي أن $q='ABDELKADE*'$ بدلا من $q='ABDEL8ADE*'$
- استبدال الحرف رقم 10 الذي هو * بالحرف R و عليه يصبح محتوى المتغيرة q هو $ABDELKADER$ بدلا من $ABDELKADE*$ أي أن $q='ABDELKADER'$ بدلا من $q='ABDELKADE*'$

4. التغييرات على المتغيرة $l='HAYETTE'$

- استبدال الحرف رقم 1 الذي هو H بالحرف R
- استبدال الحرف رقم 3 الذي هو Y بالرمز %
- استبدال الحرف رقم 6 الذي هو T بالرمز §

و عليه يصبح محتوى المتغيرة l هو $RAY\%T\S E'$ بدلا من $HAYETTE$ أي أن $l='RAY\%T\S E''$ بدلا من $l='HAYETTE'$

من خلال المثال 01.06 أعلاه اتضح و تبين

يتم إجراء أي تغيير على أي محتوى متغيرة من النوع سلسلة أحرف وفقا لطريقة الكتابة (La Syntaxe) التالية:

La Variable Chaine De Caractères [Index] := ' Caractère ' ;

حيث أن:

- La Variable Chaine De Caractères: تمثل المتغيرة سلسلة أحرف المراد إجراء عليها التغيير.
- **[ ]** : Deux Crochets.
- Indexe : يمثل رتبة أو رقم الحرف المراد استبداله.

- Deux Quotes : ‘ ‘
- Caractère : يمثل الحرف المراد الاستبدال به.

بغرض التوضيح أكثر نأخذ المثال التالي:

مثـــال 02.06:

لتكن لدينا المتغيرة سلسلة أحرف x و التي تحتوي على المعطية سلسلة أحرف *SOUGUEUR* أي: $x='SOUGUEUR$. حيث نود إجراء التغيير التالي على محتوى المتغيرة x كما يلي:

- استبدال الحرف رقم 1 بالحرف F .

يتم ذالك وفقا للكتابة (La Syntaxe) التالية :

x **[1] := ‘F‘ ;**

و عليه يصبح محتوى المتغير x هو *FOUGUEUR* أي أن $x='FOUGUEUR$

البرنامج الذي يسمح بإدخال قيمة المتغيرة x التي هي *SOUGUEUR* و إجراء التغيير هو التالي:

```
program XXXX ;
var  x : string ;
begin
     x:='SOUGUEUR';
     x[1]:='F' ;
     write (x);
     readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
FOUGUEUR_
```

مثـــال 03.06:

لتكن لدينا المتغيرة سلسلة أحرف $y1$ و التي تحتوي على المعطية سلسلة أحرف *UNIVERSITE* أي: $y1='UNIVERSITE'$. حيث نود إجراء التغييرات التالية:

- استبدال الحرف رقم 1 بالحرف T .

- استبدال الحرف رقم 3 بالحرف G .
- استبدال الحرف رقم 4 بالرمز = .
- استبدال الحرف رقم 5 بالرمز ? .
- استبدال الحرف رقم 6 بالرقم 8 .
- استبدال الحرف رقم 7 بالرمز + .
- استبدال الحرف رقم 8 بالرمز * .
- استبدال الحرف رقم 10 بالحرف Z .

يتم ذالك وفقا للكتابة (La Syntaxe) التالية :

```
y1 [ 1 ] := 'T' ;
y1 [ 3 ] := 'G' ;
y1 [ 4 ] := '=' ;
y1 [ 5 ] := '?' ;
y1 [ 6 ] := '8' ;
y1 [ 7 ] := '+' ;
y1 [ 8 ] := '*' ;
y1 [ 10 ] := 'Z' ;
```

و عليه يصبح محتوى المتغير $y1$ هو $TNG=?8+*TZ$ أي أن $y1='TNG=?8+*TZ'$

البرنامج الذي يسمح بإدخال قيمة المتغيرة $y1$ التي هي $UNIVERSITE$ و إجراء التغيير هو التالي:

```
program XXXX ;
var  y1 : string ;
begin
     y1:='UNIVERSITE';
     y1 [ 1 ] := 'T' ;
     y1 [ 3 ] := 'G' ;
     y1 [ 4 ] := '=' ;
     y1 [ 5 ] := '?' ;
     y1 [ 6 ] := '8' ;
     y1 [ 7 ] := '+' ;
     y1 [ 8 ] := '*' ;
     y1 [ 10 ] := 'Z' ;
     write ( y1);
     readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
TNG= ?8+*TZ
```

03.06 – ربـط المعطيـات من النـوع سلسلـة أحـرف:

La Concaténation Des Données Chaine De Caractères :

يمكن أن نربط أو أن نجمع بين محتوى متغيرتين من النوع سلسلة أحرف، كما يمكن الربط أو الجمع (La Concaténation) بين أكثر من محتوى متغيرتين أي بين محتوى العديد من المتغيرات من النوع سلسلة أحرف. بغرض توضيح ماذا نعني بعملية الربط و ليس كيف تتم عملية الربط، نأخذ المثال التالي:

مثـــال 04.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$$x = 'BONNE'$$
$$y = 'ANNEE'$$

لنفرض أننا نريد الربط أو الجمع بين محتوى المتغيرتين x و y أي الربط أو الجمع بين $BONNE$ و $ANNEE$.

نتيجة عملية الربط أو الجمع هي $BONNEANNEE$ و التي سوف يتم تخزينها في متغيرة أخرى أي ثالثة و لتكن z من نفس النوع أي من النوع سلسلة أحرف.

مثـــال 05.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$$b = 'DAIRA'$$
$$c = 'DE'$$
$$w = 'SOUGUEUR'$$

لنفرض أننا نريد الربط و الجمع بين محتوى الثلاث متغيرات b و c و w أي نربط و نجمع بين $DAIRA$ و DE و $SOUGUEUR$.

نتيجة عملية الربط أو الجمع هي *DAIRADESOUGUEUR* و التي سوف يتم تخزينها في متغيرة أخرى أي رابعة و لتكن d من النوع سلسلة أحرف.

ملاحظـة 01.06 :

ما يلاحظ أو ما يقال عن نتيجة عملية الربط أو الجمع هو ما يلي:

- نتيجة عملية ربط و جمع محتوى متغيرتين من النوع سلسلة أحرف أو أكثر يتم توجيهه و تخزينه في متغيرة أخرى من نفس النوع أي من النوع سلسلة أحرف.
- نتيجة عملية الربط و الجمع ليس لها أي معنى، بعبارة أخرى لا يمكن قراءتها رغم أن محتوى المتغيرات التي تم استخدامها في عملية الربط له معنى و يمكن قراءته. مثل ما هو الحال بالنسبة للمثال 04.04 حيث نتيجة عملية الربط هي: *BONNEANNEE* و التي لا يمكن قراءتها إلا إذا تم إضافة فراغ (Espace) داخل هذه النتيجة بين *BONNE* التي تمثل محتوى المتغيرة x و *ANNEE* التي تمثل محتوى المتغيرة y حيث نحصل على *BONNE ANNEE*. يتم إضافة هذا الفراغ (Espace) عن طريق الاستعانة بمتغيرة أخرى إضافة إلى المتغيرتين x و y، هذه المتغيرة من النوع سلسلة أحرف و لتكن w محتواها عبارة أو يساوي فراغ واحد (Un Espace) أي أن $w=' '$ ثم نقوم بعملية الربط و الجمع بين المتغيرات x و y و w و مثلما يوضحه المثال 06.04 أدناه

مثـــال 06.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$$x='BONNE'$$
$$w=' '$$
$$y='ANNEE'$$

لنفرض أننا نريد الربط أو الجمع بين محتوى الثلاث متغيرات x و y و w أي الربط أو الجمع بين *BONNE* و ' ' و *ANNEE*.

نتيجة عملية الربط أو الجمع هي *BONNE ANNEE* و التي سوف يتم تخزينها في متغيرة أخرى أي رابعة و لتكن z من نفس النوع أي من النوع سلسلة أحرف. نلاحظ أن نتيجة عملية الربط و الجمع لها معنى أي يمكن قراءتها.

بالنسبة للمثال أعلاه 06.06 نقوم بإضافة متغيرتين $e1$ و $e2$ من النوع سلسلة أحرف كل منهما تساوي فراغ واحد أي $e1 = ' '$ و $e2 = ' '$ كما هو موضح في المثال 07.04 أدناه

مثـــال 07.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$$b = 'DAIRA'$$

$$e1 = ' '$$

$$c = 'DE'$$

$$e2 = ' '$$

$$w = 'SOUGUEUR'$$

لنفرض أننا نريد الربط و الجمع بين محتوى الخمس متغيرات b و $e1$ و c و $e2$ و w أي نربط و نجمع بين $DAIRA$ و ' ' و DE و ' ' و $SOUGUEUR$.

نتيجة عملية الربط أو الجمع هي $DAIRA\ \ DE\ \ SOUGUEUR$ و التي سوف يتم تخزينها في متغيرة أخرى أي سادسة و لتكن d من النوع سلسلة أحرف.

بعدما تم التطرق إلى ماذا تعني عملية الربط و الجمع سوف يتم التطرق إلى كيف تتم هذه العملية. تتم عملية ربط و جمع محتوى المتغيرات من النوع سلسلة أحرف باستخدام ما يلي:

1. علامة الجمع + L'opérateur d'adition
2. الدالة **concat ( )** L'instruction (Fonction)

01.03.06 – الربـط باستخـدام علامـة الجمـع +:

La Concaténation Par L'opérateur D'adition +:

طريق الكتابة التي تسمح بربط محتوى متغيرات من النوع سلسلة أحرف هي التالية:

```
Variable chaine := chaine_1 + chaine_2 + chaine_3 + …….+ chaine_n
```

حيث أن:

- Variable chaine: تمثل متغيرة من النوع سلسلة أحرف التي يتم فيها تخزين نتيجة عملية ربط و جمع محتوى المتغيرات: chaine_1,chaine_2, chaine_3, ……,chaine_n.

- chaine_1,chaine_2, chaine_3, ……,chaine_n : تمثل المتغيرات من النوع سلسلة أحرف التي نريد ربط و جمع محتواها.
- + : تمثل علامة الجمع يسمح بربط محتوى المتغيرات:

chaine_1 , chaine_2 , chaine_3 , …… , chaine_n

- **:=** : يسمح بتوجيه و تخزين نتيجة عملية الربط داخل المتغيرة Variable chaine.

بغرض التوضيح كيفية استخدام علامة الجمع + في عملية الربط نأخذ المثال التالي:

مثـــال 08.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$x_1 = 'UNIVERSITE'$

$a = 'DE'$

$b_1 = 'TIARET'$

عملية الربط و الجمع بين محتوى الثلاث متغيرات x_1 و a و b_1 تتم باستخدام معامل الجمع + وفقا للكتابة التالية:

$$z := x_1 + a + b_1 ;$$

نتيجة عملية الربط و الجمع هي *UNIVERSITEDETIARET* و التي سوف يتم تخزينها في متغيرة أخرى أي رابعة و لتكن z من نفس النوع أي من النوع سلسلة أحرف.

البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var  x_1 , a , b_1 , z : string ;
begin
        x_1:='UNIVERSITE';
        a:='DE';
        b_1:='TIARET';
        z:= x_1 + a + b_1 ;
        writeln (z);
        readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

ما يقال عن نتيجة عملية الربط و التي هي *UNIVERSITEDETIARET* أن ليس لها أي معنى، بعبارة أخرى لا يمكن قراءتها رغم أن محتوى المتغيرات x_1 و a و b_1 التي تم استخدامها في عملية الربط له معنى و يمكن قراءته. بغرض تجاوز هذا المشكل نقوم بإضافة أو الاستعانة بمتغيرتين إضافيتين esp_1 و $esp2$ من النوع سلسلة أحرف كل منهما تساوي فراغ واحد أي $esp_1=' '$ و $esp_2=' '$ لنقوم بعد ذلك بالربط بين الخمس متغيرات x_1 و esp_1 و a و esp_2 و b_1 باستخدام معامل الجمع + وفقا للكتابة التالية:

$$z := x_1 + esp_1 + a + esp_2 + b_1 ;$$

نتيجة عملية الربط و الجمع هي *UNIVERSITE DE TIARET* و التي سوف يتم تخزينها في متغيرة أخرى أي رابعة و لتكن z من نفس النوع أي من النوع سلسلة أحرف.

البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var  x_1 , a , b_1 , z : string ;
begin
        x_1:='UNIVERSITE';
        a:='DE';
        b_1:='TIARET';
        esp_1:=' ' ;
        esp_2:=' ' ;
        z:= x_1 + esp_1 + a + esp_2 + b_1 ;
        writeln (z);
        readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

UNIVERSITE DE TIARET

مثـــال 09.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$ch1 = 'LA'$

$ch2 = 'VILLE'$

$ch3 = 'DE'$

$ch4 = 'TIARET'$

$ch5 = 'EST'$

$ch6 = 'BELLE'$

$esp = ' \; '$

نريد الربط بين محتوى المتغيرات التالية:

- $ch1$ و $ch6$ و تخزين النتيجة في المتغيرة $ch16$
- $ch6$ و $ch1$ و تخزين النتيجة في المتغيرة $ch61$
- $ch6$ و $ch6$ و تخزين النتيجة في المتغيرة $ch66$
- $ch2$ و $ch4$ و $ch2$ تخزين النتيجة في المتغيرة $ch242$
- $ch5$ و $ch2$ و تخزين النتيجة المتغيرة $ch52$
- $ch1$ و $ch2$ و $ch6$ و تخزين النتيجة في المتغيرة $ch126$
- $ch6$ و $ch2$ و $ch5$ تخزين النتيجة في المتغيرة $ch625$
- $ch4$ و $ch1$ و $ch6$ و $ch3$ و تخزين النتيجة في المتغيرة $ch4163$
- $ch6$ و $ch5$ و $ch4$ و $ch3$ و $ch2$ و $ch1$ تخزين النتيجة في المتغيرة ch
- $ch1$ و $ch2$ و $ch3$ و $ch4$ و $ch5$ و $ch6$ تخزين النتيجة في المتغيرة chh
- $ch1$ و esp و $ch2$ و esp و $ch3$ و esp و $ch4$ و esp و $ch5$ و esp و $ch6$ تخزين النتيجة في المتغيرة $chhh$

عملية الربط بين محتوى المتغيرات أعلاه تتم باستخدام معامل الجمع + وفقا للكتابات التالية:

```
ch16 := ch1 + ch6 ;
ch61 := ch6 + ch1 ;
ch66 := ch6 + ch6 ;
ch242 := ch2 + ch4 + ch2 ;
ch52 := ch5 + ch2 ;
ch126 := ch1 + ch2 + ch6 ;
ch625 := ch6 + ch2 + ch5 ;
ch4163 := ch4 + ch1 + ch6 + ch3 ;
ch := ch6 + ch5 + ch4 + ch3 + ch2 + ch1 ;
chh := ch1 + ch2 + ch3 + ch4 + ch5 + ch6 ;
chhh := ch1+esp+ch2+esp+ch3+esp+ch4+esp+ch5+esp+ch6 ;
```

البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var  ch1,ch2,ch3,ch4,ch5,ch5,ch6,esp : string ;
     ch16,ch61,ch66,ch242,ch52,ch126,ch625,ch4163 : string ;
     ch,chh,chhh : string ;
begin
     ch1='LA' ;
     ch2='VILLE' ;
     ch3='DE' ;
     ch4='TIARET' ;
     ch5='EST' ;
     ch6='BELLE' ;
     esp=' ' ;

     ch16 := ch1 + ch6 ;
     ch61 := ch6 + ch1 ;
     ch66 := ch6 + ch6 ;
     ch242 := ch2 + ch4 + ch2 ;
     ch52 := ch5 + ch2 ;
     ch126 := ch1 + ch2 + ch6 ;
     ch625 := ch6 + ch2 + ch5 ;
     ch4163 := ch4 + ch1 + ch6 + ch3 ;
     ch := ch6 + ch5 + ch4 + ch3 + ch2 + ch1 ;
```

```
    chh := ch1 + ch2 + ch3 + ch4 + ch5 + ch6 ;
    chhh := ch1+esp+ch2+esp+ch3+esp+ch4+esp+ch5+esp+ch6 ;
    writeln (ch16) ;
    writeln (ch61) ;
    writeln (ch66) ;
    writeln (ch242) ;
    writeln (ch52) ;
    writeln (ch126) ;
    writeln (ch625) ;
    writeln (ch4163) ;
    writeln (ch) ;
    writeln (chh) ;
    writeln (chhh) ;
    readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
LABELLE
BELLELA
BELLEBELLE
VILLETIARETVILLE
ESTVILLE
LAVILLEBELLE
BELLEVILLEEST
TIARETLABELLEDE
BELLEESTTIARETDEVILLELA
LAVILLEDETIARETESTBELLE
LA VILLE DE TIARET EST BELLE
```

ملاحظـة 02.06 :

مثلما يمكن ربط و جمع **محتوى** متغيرات من النوع سلسلة أحرف، يمكن كذلك ربط **معطيات** من النوع سلسلة أحرف دون اللجوء إلى إدخال هذه المعطيات في متغيرات من النوع سلسلة أحرف ثم نقوم بعد ذلك بربطها. مثلما يوضحه المثال 10.06 أدناه

مثـــال 10.06:

لتكن لدينا المعطيات من النوع سلسلة أحرف التالية:

SETTI

HAMID

ENSEIGNANT

نريد ربط هذه المعطيات و تخزين نتيجة عملية الربط في متغيرة من النوع سلسلة أحرف *k* يتم ذلك وفقا لطريقة الكتابة التالية:

k := '*SETTI*' + '*HAMID*' + '*ENSEIGNANT*' ;

البرنامج الذي يسمح بعملية ربط المعطيات أعلاه و تخزين نتيجة عملية الربط في المتغيرة *k* ثم نشر محتوى *k* هو التالي:

```
program Concatenation ;
var  k : string ;
begin
      k := 'SETTI' + 'HAMID' + 'ENSEIGNAT' ;
      writeln (k);
      readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
SETTIHAMIDENSEIGNANT
```

حتى يكون محتوى المتغيرة له معنى و مقروء نقوم بإدراج الفراغات التالية:

- فراغ واحد بين *SETTI* و *HAMID*
- فراغ واحد بين *HAMID* و *ENSEIGNANT*

يتم ذلك وفقا لطريقة الكتابة التالية:

k := '*SETTI*' + ' ' + '*HAMID*' + ' ' + '*ENSEIGNANT*' ;

و عليه يصبح البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var  k : string ;
begin

  k := 'SETTI' + '  ' + 'HAMID' + '  ' + 'ENSEIGNAT' ;

  writeln (k);
  readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
SETTI  HAMID  ENSEIGNANT
```

مما سبق التطرق إليه أعلاه يمكن استنتاج طريقة الكتابة الثانية (La Deuxième Syntaxe) لعملية الربط و هي التالية:

Variable chaine **:=** 'ch_1'+'ch_2'+'ch_3' +'ch_4' + …….+ 'ch_n'

حيث أن:

- Variable chaine: تمثل متغيرة من النوع سلسلة أحرف التي يتم فيها تخزين نتيجة عملية ربط المعطيات من النوع سلسلة أحرف : ch_1,ch_2, ch_3, ch_4, ……,ch_n.
- ch_1,ch_2, ch_3, ch_4, ……,ch_n: تمثل المعطيات من النوع سلسلة أحرف التي نريد ربطها.
- ' ' : تمثل و التي يتم بداخلها وضع المعطية من نوع سلسلة أحرف
- + : تمثل علامة الجمع يسمح بربط المعطيات ch_1,ch_2, ch_3, ch_4, ……,ch_n
- **:=** : يسمح بتوجيه و تخزين نتيجة عملية الربط داخل المتغيرة Variable chaine.

ملاحظـة 03.06 :

إضافة إلى أنه يمكن ربط **محتوى** متغيرات من النوع سلسلة أحرف و ربط **معطيات** من النوع

سلسلة أحرف، فإنه يمكن الربط بين الاثنين معا أي الربط بين محتوى متغيرات و معطيات مثلما يوضحه المثال 11.06 أدناه

مثـــال 11.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

IBN KHALDOUN

UNIVERSITE

*ch*1 = '*LA*'

*ch*2 = '*WILAYA*'

*ch*3 = '*DE*'

TIARET

*ch*4 = ' '

نريد الربط بين ما يلي:

- *ch*1 و *ch*2 و *ch*3 و *TIARET* و تخزين النتيجة في المتغيرة *ch*123.
- *ch*1 و *ch*4 و *ch*2 و *ch*4 و *ch*3 و *ch*4 و *TIARET* و تخزين النتيجة في المتغيرة *ch*124.
- *UNIVERSITE* و *ch*3 و *ch*1 و *ch*2 و *ch*3 و *TIARET* و تخزين النتيجة في المتغيرة *ch*125.
- *UNIVERSITE* و *ch*4 و *ch*3 و *ch*4 و *ch*1 و *ch*4 و *ch*2 و *ch*4 و *ch*3 و *ch*4 و *TIARET* و تخزين النتيجة في المتغيرة *ch*126.

عمليات الربط أعلاه تتم باستخدام معامل الجمع + وفقا للكتابات التالية:

```
ch123 := ch1 + ch2 + ch3 + 'TIARET' ;
ch124 := ch1 + ch4 + ch2 + ch4 + ch3 + ch4 + 'TIARET' ;
ch125 := 'UNIVERSITE' + ch3 + ch1 + ch2 + ch3 + 'TIARET' ;
ch126:= 'UNIVERSITE'+ch4+ch3+ch4+ch1+ch4+ch2+ch4+ch3+ch4+'TIARET' ;
ch127 := 'UNIVERSITE' + 'IBN KHALDOUN + 'TIARET' ;
```

البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var ch1,ch2,ch3,ch4,ch123,ch124,ch125,ch126,ch127,ch128 : string ;
begin
```

```
ch4 = ' '
ch1 = 'LA' ;
ch2 = 'WILAYA' ;
ch3 = 'DE' ;
ch123 := ch1 + ch2 + ch3 + 'TIARET' ;
ch124 := ch1 + ch4 + ch2 + ch4 + ch3 + ch4 + 'TIARET' ;
ch125 := UNIVERSITE + ch3 + ch1 + ch2 + ch3 + 'TIARET' ;
ch126 := UNIVERSITE + ch4 + ch3 + ch4 + ch1 + ch4 + ch2 + ch4 + ch3 + ch4 + 'TIARET' ;
ch127 := 'UNIVERSITE' + 'IBN KHALDOUN' + 'TIARET' ;
ch128 := 'UNIVERSITE' + ch4 + 'IBN KHALDOUN' + ch4 + 'TIARET' ;
 writeln (ch123) ;
 writeln (ch124) ;
 writeln (ch125) ;
 writeln (ch126) ;
 writeln (ch127) ;
 writeln (ch128) ;
 readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
LAWILAYADETIARET
LA WILAYA DE TIARET
UNIVERSITEDELAWILAYADETIARET
UNIVERSITE DE LA WILAYA DE TIARET
UNIVERSITEIBN KHALDOUNTIARET
UNIVERSITE IBN KHALDOUN TIARET
```

02.03.06 – الربـط باستخـدام الدالـة Concat( ):

La Concaténation Par L'instruction (La Fonction) Concat:

إضافة إلى معامل الجمع + الذي يقوم بعملية ربط و جمع المعطيات من النوع سلسلة أحرف، هناك دالة تقوم كذلك بعملية الربط. هذه الدالة هي:

concat( )

هذه الدالة تقوم بعملية الربط وفقا لطريقة الكتابة (La Syntaxe) التالية:

Variable chaine **:= concat(** chaine_1**,**chaine_2**,**chaine_3**,**…….**,**chaine_n **)**

حيث أن:

- Variable chaine: تمثل متغيرة من النوع سلسلة أحرف التي يتم فيها تخزين نتيجة عملية ربط محتوى المتغيرات: chaine_1,chaine_2, chaine_3, ……,chaine_n.
- chaine_1**,**chaine_2, chaine_3**,** ……,chaine_n : تمثل المتغيرات من النوع سلسلة أحرف التي نريد ربط محتواها.
- **Concat( )** : تمثل الدالة التي تقوم بربط محتوى المتغيرات:
  chaine_1 **,** chaine_2 , chaine_3 **,** …… , chaine_n
- **:=** : يسمح بتوجيه و تخزين نتيجة عملية الربط داخل المتغيرة Variable chaine.

بغرض التوضيح كيفية استخدام الدالة **concat( )** في عملية الربط نأخذ المثال التالي:

مثـــال 12.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$x = DATE$

$y = NAISSANCE$

$z = DE$

$w = ' \ '$

عملية الربط و الجمع بين محتوى الثلاث متغيرات x و z و y تتم باستخدام الدالة **concat( )** وفقا لطريقة الكتابة التالية:

a := concat(x , z , y) ;

نتيجة عملية الربط و الجمع هي *DATEDENAISSANCE* و التي سوف يتم تخزينها في متغيرة أخرى أي رابعة و لتكن k من نفس النوع أي من النوع سلسلة أحرف.

البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var  x , y , z , w a : string ;
begin
        x:='DATE ';
        y:='NAISSANCE ';
        z:='DE' ;
        w:=' ' ;
        a := concat( x , z , y ) ;
        writeln (a);
        readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
DATEDENAISSANCE
```

ما يقال عن نتيجة عملية الربط و التي هي *DATEDENAISSANCE* أن ليس لها أي معنى، بعبارة أخرى لا يمكن قراءتها رغم أن محتوى المتغيرات x و z و y التي تم استخدامها في عملية الربط له معنى و يمكن قراءته. بغرض تجاوز هذا المشكل نقوم بالاستعانة بالمتغيرة w التي تساوي فراغ واحد لنقوم بعد ذلك بعملية الربط باستخدام الدالة **( )concat** وفقا لطريقة الكتابة التالية:

```
a := concat( x , w , z , w , y ) ;
```

نتيجة عملية الربط هي *DATE DE NAISSANCE* و التي سوف يتم تخزينها في متغيرة أخرى و لتكن w من النوع سلسلة أحرف.

البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var  x , y , z , w a : string ;
begin
        x:='DATE ';
        y:='NAISSANCE ';
        z:='DE' ;
        w:=' ' ;
```

```
    a := concat( x , z , y ) ;
    writeln (a);
    readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
DATE  DE  NAISSANCE
```

مثـــال 13.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$a1 = JE$

$a2 = TE$

$a3 = SOUHAITE$

$a4 = UNE$

$a5 = BONNE$

$a6 = NUIT$

$a7 = '\ '$

نريد الربط بين محتوى المتغيرات التالية:

- $a4$ و $a6$ و تخزين النتيجة في المتغيرة $a46$
- $a4$ و $a7$ و $a6$ و تخزين النتيجة في المتغيرة $a476$
- $a6$ و $a6$ و تخزين النتيجة في المتغيرة $a66$
- $a6$ و $a7$ و $a6$ و تخزين النتيجة في المتغيرة $a676$
- $a1$ و $a2$ و $a3$ و تخزين النتيجة في المتغيرة $a123$
- $a1$ و $a7$ و $a2$ و $a7$ و $a3$ و تخزين النتيجة في المتغيرة $a17273$
- $a1$ و $a2$ و تخزين النتيجة في المتغيرة $a12$
- $a1$ و $a7$ و $a2$ و تخزين النتيجة في المتغيرة $a172$

- $a5$ و $a6$ و تخزين النتيجة في المتغيرة $a56$
- $a5$ و $a7$ و $a6$ و تخزين النتيجة في المتغيرة $a576$
- $a1$ و $a2$ و $a3$ و $a4$ و $a5$ و $a6$ و تخزين النتيجة في المتغيرة $a123456$
- $a1$ و $a7$ و $a2$ و $a7$ و $a3$ و $a7$ و $a4$ و $a7$ و $a5$ و $a7$ و $a6$ و تخزين النتيجة في المتغيرة a

عملية الربط بين محتوى المتغيرات أعلاه تتم باستخدام الدالة **concat( )** وفقا للكتابات التالية:

$a46$ **:= Concat (** $a4, a6$ **) ;**

$a476$ **:= Concat (** $a4, a7, a6$ **) ;**

$a66$ **:= Concat (** $a6, a6$ **) ;**

$a676$ **:= Concat (** $a6, a7, a6$ **) ;**

$a123$ **:= Concat (** $a1, a2, a3$ **) ;**

$a17273$ **:= Concat (** $a1, a7, a2, a7, a3$ **) ;**

$a12$ **:= Concat (** $a1, a2$ **) ;**

$a172$ **:= Concat (** $a1, a7, a2$ **) ;**

$a56$ **:= Concat (** $a5, a6$ **) ;**

$a576$ **:= Concat (** $a5, a7, a6$ **) ;**

$a123456$ **:= Concat (** $a1, a2, a3, a4, a5, a6$ **) ;**

a **:= Concat (** $a1, a7, a2, a7, a3, a7, a4, a7, a5, a7, a6$ **) ;**

البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var  a1,a2,a3,a4,a5,a6,a7 : string ;
     a46,a476,a66,a676    : string ;
     a123,a17273,a12,a172 : string ;
     a56,a576,a123456,a   : string ;
 begin
        a1:='JE';
        a2:='TE';
```

```
a3:='SOUHAITE' ;
a4:='UNE';
a5:='BONNE';
a6:='NUIT' ;
a7:=' ' ;
a46:= concat( a4,a6) ;
a476:= concat( a4,a7,a6) ;
a66:= concat( a6,a6 ) ;
a676:= concat( a6,a7,a6 ) ;
a123:= concat( a1,a2,a3 ) ;
a17273:= concat( a1,a7,a2,a7,a3 ) ;
a12:= concat( a1,a2 ) ;
a172:= concat( a1,a7,a2 ) ;
a56:= concat( a5,a6) ;
a576:= concat( a5,a7,a6) ;
a123456:= concat( a1,a2,a3,a4,a5,a6 ) ;
a:= concat( a1,a7,a2,a7,a3,a7,a4,a7,a5,a7,a6 ) ;
writeln (a46);
 writeln (a476);
 writeln (a66);
 writeln (a676);
 writeln (a123);
 writeln (a17273);
 writeln (a12);
 writeln (a172);
 writeln (a56);
 writeln (a576);
 writeln (a123456);
 writeln (a);
 readln ;
```

```
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
UNENUIT
UNE  NUIT
NUITNUIT
NUIT  NUIT
JETESOUHAITE
JE  TE  SOUHAITE
JETE
JE  TE
BONNENUIT
BONNE  NUIT
JETESOUHAITEUNEBONNENUIT
JE  TE  SOUHAITE  UNE  BONNE  NUIT
```

ملاحظـة 04.06 :

مثلما يمكن للدالة **( )concat** ربط **محتوى** متغيرات من النوع سلسلة أحرف، يمكن لها كذلك ربط **معطيات** من النوع سلسلة أحرف دون اللجوء إلى إدخال هذه المعطيات في متغيرات من النوع سلسلة أحرف ثم نقوم بعد ذلك بربطها. مثلما يوضحه المثال 14.04 أدناه

مثـــال 14.06:

لتكن لدينا المعطيات من النوع سلسلة أحرف التالية:

2011

/

2012

ANNEE

UNIVERSITAIRE

نريد ربط هذه المعطيات و تخزين نتيجة عملية الربط في متغيرة من النوع سلسلة أحرف k يتم ذلك باستخدام الدالة **( )concat** وفقا لطريقة الكتابة التالية:

k := **concat(**'*ANNEE*' **,** '*UNIVERSITAIRE*' **,** 2011**,** '/'**,** '2012' **) ;**

البرنامج الذي يسمح بعملية ربط المعطيات أعلاه و تخزين نتيجة عملية الربط في المتغيرة *k* ثم نشر محتوى *k* هو التالي:

```
program Concatenation ;
var  k : string ;
begin
     k :=concat('ANNEE','UNIVERSITAIRE',2011,'/','2012' ) ;
     writeln (k);
     readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
ANNEEUNIVERSITAIRE2011/2012
```

حتى يكون محتوى المتغيرة له معنى و مقروء نقوم بإدراج الفراغات التالية:

- فراغ واحد بين *ANNEE* و *UNIVERSITAIRE*
- فراغ واحد بين *UNIVERSITAIRE* و 2011

يتم ذلك وفقا لطريقة الكتابة التالية:

```
k :=concat('ANNEE',' ','UNIVERSITAIRE',' ',2011,'/','2012' ) ;
```

و عليه يصبح البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var  k : string ;
begin
   k :=concat('ANNEE',' ','UNIVERSITAIRE',' ',2011,'/','2012' ) ;

    writeln (k);
 readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

ANNEE UNIVERSITAIRE 2011/2012

مما سبق التطرق إليه أعلاه يمكن استنتاج طريقة الكتابة الثانية (La Deuxième Syntaxe) لعملية الربط باستخدام الدالة **( )concat** :

Variable chaine **:= concat(** 'ch_1'**,**'ch_2'**,**'ch_3' **,**'ch_4' **,** …….**,** 'ch_n' **)**

حيث أن:

- Variable chaine: تمثل متغيرة من النوع سلسلة أحرف التي يتم فيها تخزين نتيجة عملية ربط المعطيات من النوع سلسلة أحرف : ch_1,ch_2, ch_3, ch_4, …….,ch_n.
- ch_1,ch_2, ch_3, ch_4, …….,ch_n: تمثل المعطيات من النوع سلسلة أحرف التي نريد ربطها.
- ' ' : تمثل و التي يتم بداخلها وضع المعطية من نوع سلسلة أحرف
- **( )Concat +** : تمثل الدالة التي تقوم بربط محتوى المتغيرات:
  ch_1,ch_2, ch_3, ch_4, …….,ch_n
- **:=** : يسمح بتوجيه و تخزين نتيجة عملية الربط داخل المتغيرة Variable chaine.

ملاحظـة 05.06 :

إضافة إلى أن الدالة **( )concat** يمكن لها ربط **محتوى** متغيرات من النوع سلسلة أحرف و ربط **معطيات** من النوع سلسلة أحرف، فإنه يمكن لها الربط بين الاثنين معا أي الربط بين محتوى متغيرات و معطيات مثلما يوضحه المثال 15.06 أدناه

مثـــال 15.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

*ch*1 = *PROGRAMMTION*

LANGAGE

$ch2 = EN$

$ch3 = PASCAL$

$ch5 = 'DE'$

$LIVRE$
$ch4 = ' '$

نريد الربط بين ما يلي:

- $LANGAGE$ و $ch3$ و تخزين النتيجة في المتغيرة $ch03$.
- $LANGAGE$ و $ch4$ و $ch3$ و تخزين النتيجة في المتغيرة $ch043$.
- $LIVRE$ و $LANGAGE$ و $ch3$ و تخزين النتيجة في المتغيرة $ch003$.
- $LIVRE$ و $ch4$ و $LANGAGE$ و $ch4$ و $ch3$ و تخزين النتيجة في المتغيرة $ch04043$.
- $LIVRE$ و $ch5$ و $ch1$ و $ch2$ و $LANGAGE$ و $ch3$ و تخزين النتيجة في المتغيرة ch.
- $LIVRE$ و $ch4$ و $ch5$ و $ch4$ و $ch1$ و $ch4$ و $ch2$ و $ch4$ و $LANGAGE$ و $ch4$ و $ch3$ و تخزين النتيجة في المتغيرة chh.

عمليات الربط أعلاه تتم باستخدام الدالة **concat( )** وفقا للكتابات التالية:

```
ch03:= concat( 'LANGAGE' , ch3 ) ;
ch043:= concat( 'LANGAGE' , ch4 , ch3 ) ;
ch003:= concat( 'LIVRE' , 'LANGAGE' , ch3 ) ;
ch04043:= concat( 'LIVRE' , ch4 , 'LANGAGE' , ch4 , ch3 ) ;
ch:= concat( 'LIVRE' , ch5 , ch1 , ch2 , 'LANGAGE' , ch3 ) ;
chh:=concat('LIVRE',ch4,ch5,ch4,ch1,ch4,ch2,ch4,'LANGAGE',ch4,ch3) ;
```

البرنامج الذي يسمح بذلك هو التالي:

```
program Concatenation ;
var   ch1,ch2,ch3,ch4,ch5,ch,chh : string ;
      ch03,ch043,ch003,ch04043   : string ;
 begin
    ch1:='PROGRAMMATION';
    ch2:='EN';
    ch3:='PASCAL' ;
    ch4:=' ';
    ch5:='DE';
    ch03:= concat( 'LANGAGE' , ch3 ) ;
    ch043:= concat( 'LANGAGE' , ch4 , ch3 ) ;
    ch003:= concat( 'LIVRE' , 'LANGAGE' , ch3 ) ;
```

```
    ch04043:= concat( 'LIVRE' , ch4 , 'LANGAGE' , ch4 , ch3 ) ;
    ch:= concat( 'LIVRE' , ch5 , ch1 , ch2 , 'LANGAGE' , ch3 ) ;
    chh:=concat('LIVRE',ch4,ch5,ch4,ch1,ch4,ch2,ch4,'LANGAGE',ch4,ch3) ;
    writeln (ch03);
    writeln (ch043);
    writeln (ch003);
    writeln (ch04043);
    writeln (ch);
    writeln (chh);
    readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
LANGAGEPASCAL
LANGAGE PASCAL
LIVRELANGAGEPASCAL
LIVRE LANGAGE PASCAL
LIVREDEPROGRAMMATIONENLANGAGEPASCAL
LIVRE DE PROGRAMMATION EN LANGAGE PASCAL
```

04.06 – طـول معطيـة سلسلـة أحـرف: الدالـة LENGTH ( ):

La Longueur D'une Donnée Chaine De Caractères
La Fonction LENGTH ( ) :

في بعض المرات نريد معرفة طول محتوى متغيرة من النوع سلسلة أحرف أي معرفة عدد الأحرف أو الحروف التي يتكون منهم محتوى متغيرة. الدالة التي تسمح بذلك هي الدالة:

LENGTH ( )

هذه الدالة تقوم بحساب طول محتوى المتغيرة من النوع سلسلة أحرف وفقا لطريقة الكتابة (La Syntaxe) التالية:

Variable Entière **:= LENGTH (** Variable Chaine De Caractères **) ;**

حيث أن:

- Variable Entière: بما أن نتيجة عملية حساب طول متغيرة من النوع سلسلة أحرف بواسطة الدالة **( ) LENGTH** عبارة عن عدد صحيح، لذلك سوف يتم تخزين هذه النتيجة في متغيرة من النوع الصحيح و عليه فإن Variable Entière تمثل متغيرة من النوع الصحيح التي يتم فيها تخزين نتيجة عملية حساب طول المتغيرة من النوع سلسلة أحرف Variable Chaine De Caractères بواسطة الدالة ( ) LENGTH.
- Variable Chaine De Caractères : تمثل المتغيرة من النوع سلسلة أحرف التي نريد حساب طول محتواها بواسطة الدالة ( ) LENGTH.
- **( ) LENGTH** : تمثل الدالة التي تقوم بحساب طول المتغيرة من النوع سلسلة أحرف Variable Chaine De Caractères
- **:=** : يسمح بتوجيه و تخزين نتيجة عملية حساب الطول داخل المتغيرة Variable Entière.

بغرض توضيح كيفية استخدام الدالة **( ) LENGTH** في عملية حساب الطول نأخذ المثال التالي:

مثـــال 16.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$$x = DATEDENAISSANCE - DE * MOHAMED$$

$$y = LA\ \ WILAYA\ \ DE\ \ TIARET$$

$$z = LA\ \ VILLE\ \ DE\ \ TIARET\ \ \ TRES\ \ \ BELLE$$

يتم معرفة طول (عدد أحرف) المتغيرات من النوع سلسلة أحرف أعلاه باستخدام الدالة **( ) LENGTH** وفقا لطريقة الكتابة التالية:

```
n :=  length( x ) ;
m :=  length( y ) ;
 l :=  length( z ) ;
```

و عليه فإن البرنامج هو التالي:

```
program Longueur ;
var   x , y , z : string ;
      n , m , l : integer ;
begin
      x:='DATEDENAISSANCE-DE*MOHAMED';
      y:='LA WILAYA DE TIARET';
      z:='LA VILLE DE TIARET EST TRES BELLE' ;
      n:= length(x);
      m:= length(y);
```

```
    l:= length(z);
    writeln (n);
    writeln (m);
    writeln (l);
    readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
26
19
33
```

ملاحظـة 06.06 :

مثلما يمكن للدالة **( ) LENGTH** حساب طول **محتوى** متغيرة من النوع سلسلة أحرف، يمكن لها كذلك حساب طول **معطية** من النوع سلسلة أحرف دون اللجوء إلى إدخال هذه المعطية في متغيرة من النوع سلسلة أحرف ثم نقوم بعد ذلك بحساب طولها. يتم ذلك وفقا لطريقة الكتابة التالية:

Variable Entière **:= LENGTH (** ' Donnée Chaine De Caractères ' **) ;**

حيث أن:

- Variable Entière: تمثل متغيرة من النوع الصحيح التي يتم فيها تخزين نتيجة عملية حساب طول المعطية من النوع سلسلة أحرف Donnée Chaine De Caractère بواسطة الدالة LENGTH ( ).
- Donnée Chaine De Caractères : تمثل المعطية من النوع سلسلة أحرف التي نريد حساب طولها بواسطة الدالة LENGTH ( ).
- **( ) LENGTH** : تمثل الدالة التي تقوم بحساب طول المعطية من النوع سلسلة أحرف Donnée Chaine De Caractères .

- **:=** : يسمح بتوجيه و تخزين نتيجة عملية حساب الطول داخل المتغيرة Variable Entière.

بغرض توضيح كيفية استخدام الدالة **LENGTH ( )** في عملية حساب طول معطية من النوع سلسلة أحرف نأخذ المثال التالي:

مثـــال 17.06:

لتكن لدينا المعطيات من النوع سلسلة أحرف التالية:

MODULE INFORMATIQUE

ANNEE UNIVERSITAIRE 2011/2012

BON03µ£v COURAGEKLM%£432*

يتم معرفة طول (عدد أحرف) المعطيات من النوع سلسلة أحرف أعلاه باستخدام الدالة **LENGTH ( )** وفقا لطريقة الكتابة التالية:

```
x := length( 'MODULE INFORMATIQUE' ) ;
y := length( 'ANNEE UNIVERSITAIRE 2011/2012') ;
z := length( 'BON03µ*£V COURAGEKLM%£432' ) ;
```

و عليه فإن البرنامج هو التالي:

```
program Longueur ;
var x , y , z : integer ;
 begin
        x := length( 'MODULE INFORMATIQUE' ) ;
        y := length( 'ANNEE UNIVERSITAIRE 2011/2012') ;
        z := length( 'BON03µ*£V COURAGEKLM%£432' ) ;
        writeln (x);
        writeln (y);
        writeln (z);
        readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
19
29
25
```

ملاحظـة 07.06 :

بما أن نتيجة عملية حساب الطول بواسطة الدالة **( ) LENGTH** عبارة عن عدد صحيح، لذلك يتم تخزين هذه النتيجة في متغيرة من النوع الصحيح ليتم بعد ذلك نشرها على الشاشة. يمكن الاستغناء عن هذه المتغيرة الصحيحة التي يتم فيها تخزين طول متغيرة من النوع سلسلة أحرف و ذلك عن طريق النشر المباشر لنتيجة الدالة **( ) LENGTH** على الشاشة كما يوضحه المثال 18.06 أدناه

مثـــال 18.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

proGRAMMATION EN PAscal

a = 2 *EME ANNEE SCIENCES COMMERCIALLES*

INStructions

b = 46587093+ =,:!*ù* *

46587093+=,:!ù*

يتم معرفة طول (عدد أحرف) المتغيرات و المعطيات من النوع سلسلة أحرف أعلاه باستخدام الدالة LENGTH ( ) دون الاستعانة بالمتغيرة الصحيحة التي يتم فيها تخزين طول المتغيرات و المعطيات و ذلك عن طريق النشر المباشر لنتيجة الدالة LENGTH ( ) على الشاشة وفقا لطريقة الكتابة التالية:

```
writeln( length('proGRAMMATION EN PASCAL') ) ;
writeln( length( a ) ) ;
writeln( length('INStructions') ) ;
writeln( length( b ) ) ;
```

و عليه فإن البرنامج هو التالي:

```
program LONGUEUR ;
var  a , b : string ;
 begin
        a :='2 EME ANNEE SCIENCES COMMERCIALLES' ;
        b :='46587093+=,:!ù*' ;
        writeln( length('proGRAMMATION EN PASCAL') ) ;
        writeln( length( a ) ) ;
        writeln( length('INStructions') ) ;
        writeln( length( b ) ) ;
        readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
23
34
12
15
```

05.06 – الدالـة ( ) COPY:

في بعض المرات نريد استخراج أو استخلاص أو مجموعة جزئية (أي مجموعة من الحروف) من معطية من النوع سلسلة أحرف. لتوضيح ذالك نأخذ المثال التالي:

مثـــال 19.06:

لتكن x متغيرة من النوع سلسلة أحرف و التي تساوي ما يلي:

$$x = LIVREDEPROGRAMMATIONENLANGAGEPASCAL$$

لنفرض أننا نريد استخلاص مجموعات جزئية من الحروف انطلاقا من محتوى المتغيرة x الذي هو: $LIVREDEPROGRAMMATIONENLANGAGEPASCAL$. من بين المجموعات الجزئية الممكن استخلاصها نذكر ما يلي:

- المجموعة الجزئية $LIVRE$: تتكون من $j = 5$ حروف أو أحرف إبتداءا من الحرف الأول $i = 1$ لمحتوى المتغيرة x.
- المجموعة الجزئية $LANGAGEPASCAL$: تتكون من $j = 13$ حروف أو أحرف إبتداءا من الحرف الثالث و العشرون $i = 23$ لمحتوى المتغيرة x.
- المجموعة الجزئية $PASCAL$: تتكون من $j = 6$ حروف أو أحرف إبتداءا من الحرف الثلاثون $i = 30$ لمحتوى المتغيرة x.
- المجموعة الجزئية $LANGAGE$: تتكون من $j = 7$ حروف أو أحرف إبتداءا من الحرف الثالث و العشرون $i = 23$ لمحتوى المتغيرة x.
- المجموعة الجزئية $PROGRAMMATIONEN$: تتكون من $j = 15$ حروف أو أحرف إبتداءا من الحرف الثامن $i = 8$ لمحتوى المتغيرة x.
- المجموعة الجزئية DE: تتكون من $j = 2$ حروف أو أحرف إبتداءا من الحرف السادس $i = 6$ لمحتوى المتغيرة x.

- المجموعة الجزئية EN : تتكون من $j=2$ حروف أو أحرف إبتداءا من الحرف السادس $i=6$ لمحتوى المتغيرة x.
- المجموعة الجزئية T: تتكون من $j=1$ حروف أو أحرف إبتداءا من الحرف السابع عشر $i=17$ لمحتوى المتغيرة x.
- المجموعة الجزئية $REDEPROGRAMMATIONENLANGAGEPASCA$: تتكون من $j=31$ حروف أو أحرف إبتداءا من الحرف الرابع $i=4$ لمحتوى المتغيرة x.

الدالة التي تقوم بعملية الاستخلاص هي الدالة COPY ( )

تقوم الدالة COPY ( ) باستخلاص (استخراج و قلع) Extraction مجموعة جزئية أي مجموعة من الأحرف أو الحروف من معطية من النوع سلسلة حروف. يتم ذالك وفقا لطريقة الكتابة (La Suntaxe) التالية:

Variable **:= COPY (** Variable Chaine De Caractères , i , j **) ;**

حيث أن:

- Variable: بما أن نتيجة عملية الاستخلاص من متغيرة من النوع سلسلة أحرف بواسطة الدالة **COPY ( )** عبارة عن معطية من النوع سلسلة أحرف فإنه سوف يتم تخزين نتيجة عملية الاستخلاص في متغيرة من النوع سلسلة أحرف و عليه فإن Variable تمثل متغيرة من النوع سلسلة التي يتم فيها تخزين نتيجة عملية الاستخلاص من المتغيرة من النوع سلسلة أحرف Variable Chaine De Caractères بواسطة الدالة COPY ( ).
- Variable Chaine De Caractères : تمثل المتغيرة من النوع سلسلة أحرف التي نريد الاستخلاص منها بواسطة الدالة COPY ( ).
- j : تمثل عدد الحروف أو الأحرف التي نريد استخلاصها.
- i : تمثل رقم الحرف الذي تبدأ منه عملية الاستخلاص.
- **COPY ( )** : تمثل الدالة التي تقوم بعملية استخلاص عدد من الحروف يساوي j إبتداءا من الحرف رقم i لمحتوى المتغيرة من النوع سلسلة أحرف Variable Chaine De Caractères
- **:=** : يسمح بتوجيه و تخزين نتيجة عملية الاستخلاص في أو داخل المتغيرة من النوع سلسلة أحرف Variable.

بغرض توضيح كيفية استخدام الدالة في عملية الاستخلاص نأخذ المثال التالي:

مثـــال 20.06:

لتكن لدينا المتغيرة من النوع سلسلة أحرف التالية:

ch = LAWILAYADETIARET

نريد إجراء مجموعة من عمليات الاستخلاص من محتوى المتغيرة *ch* كما يلي:

- استخلاص $j = 6$ أحرف ابتداءا من الحرف رقم $i = 3$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف *ch*63.

يتم ذالك باستخدام الدالة COPY ( ) وفقا للكتابة التالية:

ch63 := COPY (ch , 1 , 6) ;

- استخلاص $j = 2$ أحرف ابتداءا من الحرف رقم $i = 9$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف *ch*29.

ch29 := COPY (ch , 9 , 2) ;

- استخلاص $j = 4$ أحرف ابتداءا من الحرف رقم $i = 1$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف *ch*41.

ch41 := COPY (ch , 1 , 4) ;

- استخلاص $j = 7$ أحرف ابتداءا من الحرف رقم $i = 2$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف *ch*72.

ch72 := COPY (ch , 2 , 7) ;

- استخلاص $j = 16$ أحرف ابتداءا من الحرف رقم $i = 1$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف *ch*161.

Ch161 := COPY (ch , 1 , 16) ;

البرنامج الذي يسمح بإنجاز عمليات الاستخلاص أعلاه هو التالي:

```
program EXTRACTION ;
var  ch,ch63,ch29,ch41,ch72,ch161 : string ;
begin
      ch :='LAWILAYADETIARET' ;
      ch63 := COPY ( ch , 3 , 6 ) ;
      ch29 := COPY ( ch , 9 , 2 ) ;
      ch41 := COPY ( ch , 1 , 4 ) ;
      ch72 := COPY ( ch , 2 , 7 ) ;
      Ch161 := COPY ( ch , 1 , 16 ) ;
      writeln( ch63 ) ;
      writeln( ch29 ) ;
      writeln( ch41 ) ;
      writeln( ch72 ) ;
      writeln( ch161 ) ;
      readln ;
```

end.

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
WILAYA
DE
LAWI
AWILAYA
LAWILAYADETIARET
```

ملاحظـة 08.06 :

مثلما يمكن للدالة ( ) COPY إستخلاص مجموعة جزئية من **محتوى** متغيرة من النوع سلسلة أحرف، يمكن لها كذلك إستخلاص مجموعة جزئية من **معطية** من النوع سلسلة أحرف دون اللجوء إلى إدخال هذه المعطية في متغيرة من النوع سلسلة أحرف ثم نقوم بعد ذلك بعملية استخلاص مجموعة جزئية. يتم ذلك وفقا لطريقة الكتابة التالية:

Variable **:= COPY (** ‘ Donnée Chaine De Caractères ’ , i , j **) ;**

حيث أن:

- Variable: تمثل متغيرة من النوع سلسلة أحرف التي يتم فيها تخزين نتيجة عملية الاستخلاص من المعطية من النوع سلسلة أحرف Donnée Chaine De Caractères بواسطة الدالة COPY ( ).
- Donnée Chaine De Caractères : تمثل المعطية من النوع سلسلة أحرف التي نريد الاستخلاص منها بواسطة الدالة ( ) COPY.
- *j* : تمثل عدد الحروف أو الأحرف التي نريد استخلاصها.
- *i* : تمثل رقم الحرف الذي تبدأ منه عملية الاستخلاص.
- **COPY ( )** : تمثل الدالة التي تقوم بعملية استخلاص عدد من الحروف يساوي *j* إبتداءا من الحرف رقم *i* للمعطية من النوع سلسلة أحرف Donnée Chaine De Caractères
- **:=** : يسمح بتوجيه و تخزين نتيجة عملية الاستخلاص في أو داخل المتغيرة من النوع سلسلة أحرف Variable.

بغرض توضيح كيفية استخدام الدالة في عملية الاستخلاص نأخذ المثال التالي:

مثـــال 21.06:

لتكن لدينا المعطية النوع سلسلة أحرف التالية:

$$FARIDA12304BONNE4500$$

نريد إجراء مجموعة من عمليات الاستخلاص على المعطية أعلاه كما يلي:

- استخلاص $j=6$ أحرف ابتداءا من الحرف رقم $i=1$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch61$.

يتم ذالك باستخدام الدالة ( ) COPY وفقا للكتابة التالية:

ch61 := COPY (' *FARIDA*12304*BONNE*4500' , 1 , 6) ;

- استخلاص $j=5$ أحرف ابتداءا من الحرف رقم $i=7$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch57$.

Ch57 := COPY (' *FARIDA*12304*BONNE*4500' , 7 , 5) ;

- استخلاص $j=9$ أحرف ابتداءا من الحرف رقم $i=12$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch912$.

Ch912 := COPY (' *FARIDA*12304*BONNE*4500' , 12 , 9) ;

- استخلاص $j=4$ أحرف ابتداءا من الحرف رقم $i=17$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch417$.

Ch417 := COPY (' *FARIDA*12304*BONNE*4500' , 17 , 4) ;

- استخلاص $j=3$ أحرف ابتداءا من الحرف رقم $i=12$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch312$.

Ch312 := COPY (' *FARIDA*12304*BONNE*4500' , 12 , 3) ;

- استخلاص $j=20$ أحرف ابتداءا من الحرف رقم $i=1$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف chh.

Chh := COPY (' *FARIDA*12304*BONNE*4500' , 1 , 20) ;

البرنامج الذي يسمح بإنجاز عمليات الاستخلاص أعلاه هو التالي:

```
program EXTRACTION ;
var  chh ,ch61,ch57,ch912,ch417,ch312 : string ;
begin
    ch61 := COPY ( 'FARIDA12304BONNE4500' , 1 , 6 ) ;
    ch57 := COPY ( 'FARIDA12304BONNE4500' , 7 , 5 ) ;
```

```
    ch912 := COPY ( 'FARIDA12304BONNE4500' , 12 , 9 ) ;
    ch417 := COPY ( 'FARIDA12304BONNE4500' , 17 , 4 ) ;
    ch312 := COPY ( 'FARIDA12304BONNE4500' , 12 , 3 ) ;
    chh := COPY ( 'FARIDA12304BONNE4500' , 1 , 20 ) ;
    writeln( ch61 ) ;
    writeln( ch57 ) ;
    writeln( ch912 ) ;
    writeln( ch417 ) ;
    writeln( ch312 ) ;
    writeln( chh ) ;
    readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
FARIDA
12304
BONNE4500
4500
BON
FARIDA12304BONNE4500
```

ملاحظـة 09.06 :

بما أن نتيجة عملية الاستخلاص بواسطة الدالة **( ) COPY** عبارة عن معطية من النوع سلسلة أحرف، لذلك سوف يتم تخزين هذه النتيجة في متغيرة من النوع سلسلة أحرف ليتم بعد ذلك نشرها على الشاشة. يمكن الاستغناء عن هذه المتغيرة من النوع سلسلة أحرف التي يتم فيها تخزين نتيجة عملية الاستخلاص و ذلك عن طريق النشر المباشر لنتيجة الدالة **( ) COPY** على الشاشة كما يوضحه المثال 22.06 أدناه

مثـــال 22.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

MIcRO ORDInatEUR

$a = 1\ \ ERE\ \ AnnEE\ \ masTER$

نريد إجراء مجموعة من عمليات الاستخلاص على المعطية أعلاه كما يلي:

- استخلاص $j = 6$ أحرف ابتداءا من الحرف رقم $i = 3$ من المعطية $MIcRO\ \ ORDInatEUR$ يتم ذالك باستخدام الدالة ( ) COPY وفقا للكتابة التالية:

COPY (' *MIcRO ORDInatEUR* ' , 3 , 6) ;

و عملية نشر ما تم استخلاصه تتم بإضافة ( ) Writeln كما يلي:

Writeln (COPY (' *MIcRO ORDInatEUR* ' , 3 , 6)) ;

- استخلاص و نشر $j = 10$ أحرف ابتداءا من الحرف رقم $i = 7$ من المعطية $MIcRO\ \ ORDInatEUR$

Writeln (COPY (' *MIcRO ORDInatEUR* ' , 7 , 10)) ;

- استخلاص و نشر $j = 10$ أحرف ابتداءا من الحرف رقم $i = 1$ من محتوى المتغيرة a

Writeln (COPY (*a* , 1 , 10)) ;

- استخلاص و نشر $j = 8$ أحرف ابتداءا من الحرف رقم $i = 7$ من محتوى المتغيرة a

Writeln (COPY (*a* , 7 , 8)) ;

و عليه فإن البرنامج الذي يسمح بالاستخلاص و النشر هو التالي:

```
program EXTRACTION ;
var  a : string ;
begin
     a:='1 ERE AnnEE masTER' ;
     writeln ( COPY ( 'MIcRO ORDInatEUR' , 3 , 6 ) ) ;
     writeln ( COPY ( 'MIcRO ORDInatEUR' , 7 , 10 ) ) ;
     writeln ( COPY ( a , 1 , 10 ) ) ;
     writeln ( COPY ( a , 7 , 8 ) ) ;
 readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
cRO  OR
ORDInatEUR
1 ERE  AnnE
AnnEE  ma
```

06.06 – الدالـة DELETE ( ):

في بعض المرات نريد **حذف** (Supprimer) مجموعة جزئية (أي مجموعة من الحروف) من معطية من النوع سلسلة أحرف. لتوضيح ذالك نأخذ المثال التالي:

مثـــال 23.06:

لتكن x متغيرة من النوع سلسلة أحرف و التي تساوي ما يلي:

$$y = LIVRE55?DE098LANGAGE + \%\mu PASCAL$$

لنفرض أننا نريد حذف مجموعات جزئية من الحروف انطلاقا من محتوى المتغيرة y الذي هو: $LIVRE55?DE098LANGAGE + \%\mu PASCAL$. من بين المجموعات الجزئية الممكن حذفها نذكر ما يلي:

- المجموعة الجزئية ?55: تتكون من $j = 3$ حروف أو أحرف إبتداءا من الحرف الأول $i = 6$ لمحتوى المتغيرة x.
- المجموعة الجزئية $\%\mu PASCAL +$: تتكون من $j = 9$ حروف أو أحرف إبتداءا من الحرف الأول $i = 21$ لمحتوى المتغيرة x.
- المجموعة الجزئية 098: تتكون من $j = 3$ حروف أو أحرف إبتداءا من الحرف الأول $i = 11$ لمحتوى المتغيرة x.

الدالة التي تقوم بعملية الحذف هي الدالة DELETE ( )

تقوم الدالة DELETE ( ) بحذف Suppression مجموعة جزئية أي مجموعة من الأحرف أو الحروف من معطية من النوع سلسلة حروف. يتم ذالك وفقا لطريقة الكتابة (La Syntaxe) التالية:

DELETE (Variable Chaine De Caractères , i , j **) ;**

حيث أن:

- Variable Chaine De Caractères : تمثل المتغيرة من النوع سلسلة أحرف التي نريد حذف منها مجموعة جزئية الأحرف أو الحروف بواسطة الدالة DELETE ( ).
- j : تمثل عدد الحروف أو الأحرف التي نريد حذفها.
- i : تمثل رقم الحرف الذي تبدأ منه عملية الحذف.
- **DELETE ( )** : تمثل الدالة التي تقوم بعملية حذف عدد من الحروف يساوي j إبتداءا من الحرف رقم i لمحتوى المتغيرة من النوع سلسلة أحرف Variable Chaine De Caractères

بغرض توضيح كيفية استخدام الدالة DELETE ( ) في عملية الحذف نأخذ المثال التالي:

مثـــال 24.06:

لتكن لدينا المتغيرة من النوع سلسلة أحرف التالية:

$ch = MODULE67890INFORMATIQUE\mu *+=?\&\S$

نريد إجراء مجموعة من عمليات الحذف من محتوى المتغيرة ch كما يلي:

- حذف $j = 5$ أحرف ابتداءا من الحرف رقم $i = 7$.

يتم ذالك باستخدام الدالة ( ) DELETE وفقا للكتابة التالية:

delete (ch , 7 , 5) ;

- حذف $j = 7$ أحرف ابتداءا من الحرف رقم $i = 19$ من محتوى المتغيرة ch

delete (ch , 19 , 7) ;

البرنامج الموافق لذلك هو التالي:

```
program SUPPRESSION ;
var  ch : string ;
begin
     ch:='MODULE67890INFORMATIQUEµ*+=?&§' ;
     delete (ch , 7 , 5 ) ;
     delete (ch , 19 , 7 ) ;
     writeln ( ch ) ;
     readln ;
end.
```

عند تنفيذ البرنامج يقوم الحاسوب بما يلي:

1. عند تنفيذ التعليمة *delete (ch , 7 , 5)* يقوم الحاسوب بحذف المجموعة الجزئية المكونة من الحروف التالية 67890 و عليه يصبح المحتوى الجديد للمتغيرة ch هو التالي: $MODULEINFORMATIQUE\mu *+=?\&\S$ أي أن $ch = MODULEINFORMATIQUE\mu *+=?\&\S$ و ليس $ch = MODULE67890INFORMATIQUE\mu *+=?\&\S$
2. عند تنفيذ التعليمة *delete (ch , 19 , 7)* يقوم الحاسوب بحذف المجموعة الجزئية المكونة من الحروف التالية µ*+=?&§ و عليه يصبح المحتوى الجديد للمتغيرة ch هو التالي: $MODULEINFORMATIQUE$ أي $ch = MODULEINFORMATIQUE$
3. عند تنفيذ التعليمة *writeln (ch)* يقوم الحاسوب بنشر المحتوى الجديد للمتغيرة ch الذي هو: $MODULEINFORMATIQUE$ كما هو موضح على الشاشة أدناه:

النشر على الشاشة

MODULEINFORMATIQUE

07.06 – الدالـة ( ) INSERT:

في بعض المرات نريد **دمـج** (Insertion) محتوى متغيرة من النوع سلسلة أحرف في أو ضمن محتوى متغيرة أخرى من النوع سلسلة أحرف. لتوضيح ذالك نأخذ المثال التالي:

مثـــال 25.06:

لتكن المتغيرات من النوع سلسلة أحرف التالية:

ch1= ‘MOISJANVIER’
ch2= ‘DE’
ch3= ‘ ‘
ch4= ‘ANNEE2012’

نريد إجراء مجموعة من عمليات الدمج أو الإدماج على المتغيرات أعلاه كما يلي:

- دمج محتوى المتغيرة *ch*2 في أو ضمن محتوى المتغيرة *ch*1 ابتداءا من الحرف رقم $j = 5$ يتم ذلك كما يلي:

Insert (ch2,ch1,5) ;

يقوم الحاسوب بدمج محتوى المتغيرة *ch*2 الذي هو DE في محتوى المتغيرة *ch*1 ابتداءا من الحرف رقم $j = 5$ و بالتالي يصبح المحتوى الجديد للمتغيرة *ch*1 هو MOISDEJANVIER بدلا من MOISJANVIER أي أن ch1= ‘MOISDEJANVIER’ بدلا من ch1= ‘MOISJANVIER’

- دمج محتوى المتغيرة *ch*3 في ضمن المحتوى الجديد للمتغيرة *ch*1 ابتداءا من الحرف رقم $j = 5$ يتم ذلك كما يلي:

Insert (ch3,ch1,5) ;

يقوم الحاسوب بدمج محتوى المتغيرة *ch*3 الذي هو فراغ في المحتوى الجديد للمتغيرة *ch*1 الذي هو MOISDEJANVIER ابتداءا من الحرف رقم $j = 5$ و بالتالي يصبح المحتوى الجديد للمتغيرة

*ch*1 هو MOIS DEJANVIER بدلا من MOISDEJANVIER أي أن ch1= ‘MOIS DEJANVIER’ بدلا من ch1= ‘MOISDEJANVIER’

- دمج محتوى المتغيرة *ch*3 في أو ضمن المحتوى الجديد للمتغيرة *ch*1 ابتداءا من الحرف رقم $j = 8$

يتم ذلك كما يلي:

Insert (ch3,ch1,8) ;

يقوم الحاسوب بدمج محتوى المتغيرة *ch*3 الذي هو فراغ في المحتوى الجديد للمتغيرة *ch*1 الذي هو MOIS DEJANVIER ابتداءا من الحرف رقم $j = 8$ و بالتالي يصبح المحتوى الجديد للمتغيرة *ch*1 هو MOIS DE JANVIER بدلا من MOIS DEJANVIER أي أن ch1= ‘MOIS DE JANVIER’ بدلا من ch1= ‘MOIS DEJANVIER’

- دمج محتوى المتغيرة *ch*4 في ضمن المحتوى الجديد للمتغيرة *ch*2 ابتداءا من الحرف رقم $j = 2$ يتم ذلك كما يلي:

Insert (ch4,ch2,2) ;

يقوم الحاسوب بدمج محتوى المتغيرة *ch*4 الذي هو ANNEE2012 في محتوى المتغيرة *ch*2 الذي هو DE ابتداءا من الحرف رقم $j = 2$ و بالتالي يصبح المحتوى الجديد للمتغيرة *ch*2 هو DANNEE2012E بدلا من DE أي أن ch2= ‘DANNEE2012E’ بدلا من ch1= ‘DE’

البرنامج الذي يسمح بإنجاز ما سبق ذكره أعلاه هو التالي:

```
program INSERTION ;
var  ch1,ch2,ch3,ch4 : string ;
begin
      ch1:= 'MOISJANVIER';
      ch2:= 'DE' ;
      ch3:= ' ' ;
      ch4:= 'ANNEE2012' ;
      insert(ch2,ch1,5) ;
       insert(ch3,ch1,5);
      insert(ch3,ch1,8);
      Insert (ch4,ch2,2) ;
      writeln (ch1);
      writeln (ch2);
      readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

MOIS DE JANVIER
DANNEE2012E

08.06 – الدالـة La Fonction Ord( )

كما هو معلوم كل حرف أو رمز له شفرة (Code ASCII) فمثلا:

- شفرة الحرف *A* Le Code ASCII De La Lettre هو: 65
- شفرة الحرف *a* هو: 97
- شفرة الرمز & هو: 38
- شفرة العدد 5 هو: 53

الدالة التي تسمح بتقديم شفرة أي حرف أو رمز أو عدد هي الدالة **Ord( )** وفقا لطريقة الكتابة (La Syntaxe) التالية:

وفقا لطريقة الكتابة (La Syntaxe) التالية:

Variable De Type Entier **:= Ord (** Variable De Type Caractère **) ;**

حيث أن:

- Variable De Type Caractère: تمثل متغيرة من النوع حرف و ليس سلسلة حروف
- **Ord( )** : تمثل الدالة التي تقوم بتقديم الشفرة.
- Variable De Type Entier: تمثل متغيرة من النوع الصحيح التي يتم فيها تخزين نتيجة الدالة **Ord( )** التي تمثل الشفرة و التي هي عبارة عن عدد صحيح.

بغرض التوضيح نأخذ المثال التالي:

مثـــال 26.06:

لنفرض أننا نريد معرفة شفرة الحروف و الرموز و الأعداد التالية:

$B, C, 2, 3, /, :, !, §, m, n, <, >$

يتم ذالك باستخدام الدالة **Ord( )** كما يلي:

1. نقوم بالتصريح بـ 12 متغيرة من النوع حرف و لتكن $x1, x2, x3, ..., x12$ بغرض تخزين فيها الحروف و الرموز و الأعداد المعطاة أعلاه. يتم ذلك كما يلي:

$$Var \quad x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, x11, x12 \ : Char \ ;$$

2. نقوم بالتصريح بـ 12 متغيرة من النوع الصحيح و لتكن $y1, y2, y3, ..., y12$ بغرض تخزين فيها نتائج الدالة **Ord( )** التي سوف يتم إدخالها على المتغيرات $x1, x2, x3, ..., x12$. يتم ذلك كما يلي:

$$Var \quad y1, y2, y3, y4, y5, y6, y7, y8, y9, y10, y11, y12 \ : integer \ ;$$

3. نقوم بإدخال هذه الحروف و الرموز و الأعداد $B, C, 2, 3, /, :, !, §, m, n, <, >$ كل منها في متغيرة من النوع حرف $x1, x2, x3, ..., x12$ التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

$$x1 := 'B' \ ; \ x2 := 'C' \ ; \ x3 := '2' \ ; \ x4 := '3' \ ; \ x5 := '/' \ ; \ x6 := ':' \ ;$$

$$x7 := '!' \ ; \ x8 := '§' \ ; \ x9 := 'm' \ ; \ x10 := 'n' \ ; \ x11 := '<' \ ; \ x12 := '>'$$

4. نقوم بإدخال الدالة **Ord( )** على المتغيرات $x1, x2, x3, ..., x12$ و توجيه و تخزين النتائج في المتغيرات $y1, y2, y3, ..., y12$. يتم ذلك كما يلي:

$$y1 := Ord(x1) \ ; \ y2 := Ord(x2) \ ; \ y3 := Ord(x3) \ ; \ y4 := Ord(x4) \ ;$$

$$y5 := Ord(x5) \ ; \ y6 := Ord(x6) \ ; \ y7 := Ord(x7) \ ; \ y8 := Ord(x8) \ ;$$

$$y9 := Ord(x9) \ ; \ y10 := Ord(x10) \ ; \ y11 := Ord(x11) \ ; \ y12 := Ord(x12)$$

5. نقوم بنشر نتائج الدالة **Ord( )** التي هي مخزنة في المتغيرات $y1, y2, y3, ..., y12$. يتم ذلك كما يلي:

$$Writeln\ (y1)\ ;\ Writeln\ (y2)\ ;\ Writeln\ (y3)\ ;\ Writeln\ (y4)\ ;$$
$$Writeln\ (y5)\ ;\ Writeln\ (y6)\ ;\ Writeln\ (y7)\ ;\ Writeln\ (y8)\ ;$$
$$Writeln\ (y9)\ ;\ Writeln\ (y10)\ ;\ Writeln\ (y11)\ ;\ Writeln\ (y12)\ ;$$

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program CODE_ASCII ;
var  x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x12 : char ;
```

```
    y1,y2,y3,y4,y5,y6,y7,y8,y9,y10,y11,y12 : integer ;
begin
    x1:='B' ; x2:='C' ; x3:='2' ; x4:='3' ; x5:='/' ; x6:=':' ;
    x7:='!'; x8:='§' ; x9:='m' ; x10:='n' ; x11:='<' ; x12:='>' ;
    y1:=ord(x1); y2:=ord(x2) ; y3:=ord(x3) ; y4:=ord(x4) ;
    y5:=ord(x5) ; y6:=ord(x6) ; y7:=ord(x7) ; y8:=ord(x8) ;
    y9:=ord(x9) ; y10:=ord(x10) ; y11:=ord(x11) ; y12:=ord(x12);
    writeln(y1) ; writeln(y2) ; writeln(y3) ; writeln(y4) ;
    writeln(y5) ; writeln(y6) ; writeln(y7) ; writeln(y8) ;
    writeln(y9) ; writeln(y10) ; writeln(y11) ;
     writeln('LE  CODE  ASCII DE ', x12, 'EGALE : ', y12);
    readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
66
67
50
51
47
58
33
167
109
110
60
LE  CODE  ASCII  DE  >  EGALE : 62
```

ملاحظـة 10.06 :

مثلما يمكن للدالة **Ord( )** تقديم شفرة (Le Code ASCII) **محتوى** متغيرة من النوع حرف، يمكن لها كذلك تقديم شفرة **معطيـة** من النوع حرف دون اللجوء إلى إدخال هذه المعطية في متغيرة من النوع حرف ثم نقوم بعد ذلك بعملية تقديم شفرة. يتم ذلك وفقا لطريقة الكتابة التالية:

Variable De Type Entier **:= Ord (** Donée De Type Caractère **) ;**

حيث أن:

- Donnée De Type Caractère: تمثل معطية (و ليس متغيرة) من النوع حرف التي نريد معرفة شفرتها بواسطة الدالة **( )Ord.**

بغرض التوضيح نأخذ المثال التالي:

مثـــال 27.06:

لنفرض أننا نريد معرفة شفرة الحروف و الرموز و الأعداد التالية:

$$?,¨\$,6,7,F,f$$

يتم ذالك باستخدام الدالة **( )Ord** كما يلي:

1. نقوم بالتصريح بـ 6 متغيرات من النوع الصحيح و لتكن $y1,y2,y3,y4,y5,y6$ بغرض تخزين فيها نتائج الدالة **( )Ord** التي سوف يتم إدخالها على المعطيات $?,¨\$,6,7,F,f$. يتم ذلك كما يلي:

$$Var \quad y1,y2,y3,y4,y5,y6 : integer ;$$

2. نقوم بإدخال الدالة **( )Ord** على المعطيات $?,¨\$,6,7,F,f$. يتم ذلك كما يلي:

$$y1 := Ord('?') \ ; \ y2 := Ord('\$') \ ; \ y3 := Ord('6') \ ;$$

$$y5 := Ord('7') \ ; \ y6 := Ord('F') \ ; \ y7 := Ord('f') \ ;$$

3. نقوم بنشر نتائج الدالة **( )Ord** التي هي مخزنة في المتغيرات $y1,y2,y3,.....,y12$. يتم ذلك كما يلي:

```
Writeln ( y1 ) ; Writeln ( y2 ) ; Writeln ( y3 ) ;
Writeln ( y4 ) ; Writeln ( y5 ) ; Writeln ( y6 ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program CODE_ASCII ;
var y1,y2,y3,y4,y5,y6 : integer ;
begin
    y1:=ord('?') ; y2:=ord('$') ; y3:=ord('6') ;
    y4:=ord('7') ; y5:=ord('F') ; y6:=ord('f') ;
```

```
    writeln(y1) ; writeln(y2) ; writeln(y3) ;
    writeln(y4) ; writeln(y5) ; writeln(y6) ;
    readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
63
36
54
55
70
102
```

ملاحظـة 11.06 :

بما أن نتيجة عملية تقديم الشفرة بواسطة الدالة **( )Ord** عبارة عن معطية من النوع الصحيح، لذلك يتم تخزين هذه النتيجة في متغيرة من النوع الصحيح ليتم بعد ذلك نشرها على الشاشة. يمكن الاستغناء عن هذه المتغيرة من النوع الصحيح التي يتم فيها تخزين نتيجة عملية تقديم الشفرة بواسطة الدالة **( )Ord** و ذلك عن طريق النشر المباشر لنتيجة الدالة **( )Ord** على الشاشة كما يوضحه المثال 28.04 أدناه

مثـــال 28.06:

لتكن لدينا المتغيرات و المعطيات من النوع أحرف التالية:

$$a = 'Y'$$

$$b = 'Z'$$

$$W$$

$$X$$

يتم معرفة شفرة المعطيات و محتوى المتغيرات أعلاه باستخدام الدالة **( )Ord** دون الاستعانة بالمتغيرات من النوع الصحيح لتخزين نتيجة الدالة **( )Ord**

1. نقوم بالتصريح بالمتغيرتين a,b من النوع حرف بغرض تخزين فيهما المعطيتين الممثلتين في الحرفين Y و Z. يتم ذلك كما يلي:

Var a,b : Char ;

2. نقوم بإدخال الحرفين Y و Z منهما في متغيرة من النوع حرف a و b التي تم التصريح بها في الخطوة الأولى. يتم ذلك كما يلي:

```
a :='Y' ;
b :='Z' ;
```

3. نشر نتيجة عملية تقديم الشفرة بواسطة الدالة **Ord( )** . يتم ذلك كما يلي:

```
writeln( ord(a) )  ;
writeln( ord(b) )  ;
writeln( ord('W') )  ;
writeln( ord('X') )  ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program CODE_ASCII ;
var a,b : char ;
begin
      a:='Y';
      b:='Z';
      writeln( ord(a) )  ;
      writeln( ord(b) )  ;
      writeln( ord('W') )  ;
      writeln( ord('X') )  ;
      readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
89
90
87
88
```

09.06 – الدالـة La Fonction Chr( )

لنفرض أننا نريد نشر على الشاشة الحروف الخاصة التي لا تظهر على لوحة المفاتيح مثل $\alpha, \beta, ...$ علما أن لهذه الحروف الخاصة شفرة Code ASCII. الدالة التي تسمح بتقديم الحرف الخاص الموافق لأي شفرة هي الدالة **Chr ( )** وفقا لطريقة الكتابة (La Syntaxe) التالية:

Variable De Type De Caractère **:= Chr (** Code ASCII **) ;**

حيث أن:

- Variable De Type Caractère: تمثل متغيرة من النوع حرف يتم فيها تخزين الحرف الخاص الموافق لـ Code Ascii
- Code ASCII:
- **Chr ( )** : تمثل الدالة التي تقوم بتقديم الحرف الخاص أو الحرف الموافق لـ Code Ascii.

بغرض توضيح كيفية استخدام الدالة **Chr ( )** نأخذ المثال التالي:

مثـــال 29.06:

لنفرض أننا نريد معرفة الحروف الخاصة الموافقة للشفرات (Les Codes ASCII) التالية:

65 , 66 , 70 , 250 , 251 , 253

يتم ذالك باستخدام الدالة **Chr ( )** كما يلي:

1. نقوم بالتصريح بـ 6 متغيرة من النوع حرف و لتكن $v_1, v_2, v_3, v_4, v_5, v_6$ بغرض تخزين فيها الحروف الخاصة للموافقة للشفرات أعلاه. يتم ذلك كما يلي:

```
Var  v_1 , v_2 , v_3 , v_4 , v_5 , v_6 : Char ;
```

2. نقوم بإدخال الدالة **Chr ( )** على الشفرات 253 , 251 , 250 , 70 , 66 , 65 و توجيه و تخزين النتائج في المتغيرات $v_1, v_2, v_3, v_4, v_5, v_6$. يتم ذلك كما يلي:

```
v_1 := chr(65) ;
v_2 := chr(66) ;
v_3 := chr(70) ;
v_4 := chr(250) ;
v_5 := chr(251) ;
v_6 := chr(253) ;
```

3. نقوم بنشر نتائج الدالة **( ) Chr** التي هي مخزنة في المتغيرات $v_1, v_2, v_3, v_4, v_5, v_6$. يتم ذلك كما يلي:

```
Writeln(v_1);
Writeln(v_2);
Writeln(v_3);
Writeln(v_4);
Writeln(v_5);
Writeln(v_6);
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program LA_FONCTION_CHR ;
Var  v_1 , v_2 , v_3 , v_4 , v_5 , v_6 : Char ;
begin
v_1 := chr(65) ;
v_2 := chr(66) ;
v_3 := chr(70) ;
v_4 := chr(250) ;
v_5 := chr(251) ;
v_6 := chr(253) ;
Writeln(v_1);
Writeln(v_2);
Writeln(v_3);
Writeln(v_4);
Writeln(v_5);
Writeln(v_6);
readln;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
A
B
F
.
1
2
```

ملاحظـة 12.06 :

بما أن نتيجة عملية تقديم الحرف الخاص بشفرة معينة بواسطة الدالة **( ) Chr** عبارة عن معطية من النوع حرف، لذلك يتم تخزين هذه النتيجة في متغيرة من النوع حرف ليتم بعد ذلك نشرها على الشاشة. يمكن الاستغناء عن هذه المتغيرة من النوع حرف التي يتم فيها تخزين نتيجة عملية تقديم الحرف الخاص بواسطة الدالة **( ) Chr** و ذلك عن طريق النشر المباشر لنتيجة الدالة **( ) Chr** على الشاشة كما يوضحه المثال 30.04 أدناه

مثـــال 30.06:

لنفرض أننا نريد معرفة الحروف الخاصة الموافقة للشفرات (Les Codes ASCII) التالية:

71 , 72 , 73 , 80 , 85 , 88

يتم ذالك باستخدام الدالة **( ) Chr** كما يلي:

```
Write ln( chr(71) ) ;
Write ln( chr(72) ) ;
Write ln( chr(73) ) ;
Write ln( chr(80) ) ;
Write ln( chr(85) ) ;
Write ln( chr(88) ) ;
```

و عليه يصبح البرنامج كما يلي:

```
program LA_FONCTION_CHR ;
begin
Writeln( chr(71) ) ;
Writeln( chr(72) ) ;
Writeln( chr(73) ) ;
Writeln( chr(80) ) ;
Writeln( chr(85) ) ;
Writeln( chr(88) ) ;
readln;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

G
H
I
P
U
X

10.06 – الدالـة La Fonction Succ( )

الدالة **Succ( )** تسمح بتقديم الحرف الخاص الذي يلي حرف خاص معين وفقا لطريقة الكتابة التالية:

Variable De Type De Caractère **:= Succ (** ' Donnée De Type De Caractère ' **) ;**

حيث أن:

- Variable De Type Caractère: تمثل متغيرة من النوع حرف يتم فيها تخزين نتيجة الدالة **Succ( )** .
- **Succ( ):** تمثل الدالة التي تسمح بتقديم الحرف الخاص الذي يلي حرف خاص معين ممثل في Donnée De Type De Caractère.
- Donnée De Type De Caractère: تمثل معطية من النوع حرف التي نريد معرفة الحرف الخاص الذي يليها.

بغرض توضيح كيفية استخدام الدالة **Succ ( )** نأخذ المثال التالي:

مثـــال 31.06:

لنفرض أننا نريد معرفة الأحرف الخاصة التي تلي الأحرف الخاصة التالية:

A , X , t , x , ? , =

يتم ذالك باستخدام الدالة **Succ ( )** كما يلي:

1. نقوم بالتصريح بـ 6 متغيرة من النوع حرف و لتكن $v_1, v_2, v_3, v_4, v_5, v_6$ بغرض تخزين فيها الحروف الخاصة التي تلي الحروف الخاصة أعلاه. يتم ذلك كما يلي:

Var v_1 , v_2 , v_3 , v_4 , v_5 , v_6 : Char ;

2. نقوم بإدخال الدالة **Succ ( )** على الحروف الخاصة + , ? , x , t , X , A و توجيه و تخزين النتائج في المتغيرات $v_1, v_2, v_3, v_4, v_5, v_6$. يتم ذلك كما يلي:

```
v_1 := succ('A') ;
v_2 := succ('X') ;
v_3 := succ('t') ;
v_4 := succ('x') ;
v_5 := succ('=') ;
v_6 := succ('+') ;
```

3. نقوم بنشر نتائج الدالة **Succ ( )** التي هي مخزنة في المتغيرات $v_1, v_2, v_3, v_4, v_5, v_6$. يتم ذلك كما يلي:

```
Writeln(v_1);
Writeln(v_2);
Writeln(v_3);
Writeln(v_4);
Writeln(v_5);
Writeln(v_6);
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program LA_FONCTION_SUCC ;
Var  v_1 , v_2 , v_3 , v_4 , v_5 , v_6 : Char ;
begin
v_1 := succ('A') ;
v_2 := succ('X') ;
v_3 := succ('t') ;
v_4 := succ('x') ;
v_5 := succ('=') ;
v_6 := succ('+') ;
Writeln(v_1);
Writeln(v_2);
Writeln(v_3);
Writeln(v_4);
Writeln(v_5);
Writeln(v_6);
readln;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

11.06 – الدالـة La Fonction Pred( )

الدالة **( ) Pred** تسمح بتقديم الحرف الخاص الذي يسبق حرف خاص معين وفقا لطريقة الكتابة التالية:

Variable De Type De Caractère **:= Pred (** ' Donnée De Type De Caractère ' **) ;**

حيث أن:

- Variable De Type Caractère: تمثل متغيرة من النوع حرف يتم فيها تخزين نتيجة الدالة **( ) Pred** .
- **( ) Pred**: تمثل الدالة التي تسمح بتقديم الحرف الخاص الذي يسبق حرف خاص معين ممثل في Donnée De Type De Caractère.
- Donnée De Type De Caractère: تمثل معطية من النوع حرف التي نريد معرفة الحرف الخاص الذي يسبقه.

بغرض توضيح كيفية استخدام الدالة **( ) Pred** نأخذ المثال التالي:

مثـــال 32.06:

لنفرض أننا نريد معرفة الأحرف الخاصة التي تلي الأحرف الخاصة التالية:

B , Z , # , I , + , ?

يتم ذالك باستخدام الدالة **( ) Pred** كما يلي:

1. نقوم بالتصريح بـ 6 متغيرة من النوع حرف و لتكن $v_1, v_2, v_3, v_4, v_5, v_6$ بغرض تخزين فيها الحروف الخاصة التي تسبق الحروف الخاصة أعلاه. يتم ذلك كما يلي:

Var v_1 , v_2 , v_3 , v_4 , v_5 , v_6 : Char ;

2. نقوم بإدخال الدالة **Pred ( )** على الحروف الخاصة + , ? , x , t , X , A و توجيه و تخزين النتائج في المتغيرات $v_1, v_2, v_3, v_4, v_5, v_6$. يتم ذلك كما يلي:

```
v_1 := pred('B') ;
v_2 := pred('Z') ;
v_3 := pred('#') ;
v_4 := pred('I') ;
v_5 := pred('+') ;
v_6 := pred('?') ;
```

3. نقوم بنشر نتائج الدالة **Pred ( )** التي هي مخزنة في المتغيرات $v_1, v_2, v_3, v_4, v_5, v_6$. يتم ذلك كما يلي:

```
Writeln(v_1);
Writeln(v_2);
Writeln(v_3);
Writeln(v_4);
Writeln(v_5);
Writeln(v_6);
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program LA_FONCTION_PRED ;
Var  v_1 , v_2 , v_3 , v_4 , v_5 , v_6 : Char ;
begin
v_1 := pred('B') ;
v_2 := pred('Z') ;
v_3 := pred('#') ;
v_4 := pred('I') ;
v_5 := pred('+') ;
v_6 := pred('?') ;
Writeln(v_1);
Writeln(v_2);
Writeln(v_3);
Writeln(v_4);
Writeln(v_5);
Writeln(v_6);
readln;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

A
Y
‘’
H
*
>

تمـاريـن الفصـــل

✍ التمــريــن 01.06:

أذكر مختلف العمليات الممكن إجرائها على المعطيات من النوع سلسلة أحرف.

✍ التمــريــن 02.06:

1. ما هي التعليمة التي تسمح بتغيير محتوى متغيرة أو معطية من النوع سلسلة أحرف.
2. قدم طريقة الكتابة للتعليمة المتوصل إليها في السؤال الأول.

✍ التمــريــن 03.06:

لتكن المتغيرة من النوع سلسلة أحرف التالية:

$$n = 'Z9HRA'$$

نود إجراء التعديل التالي: استبدال الحرف رقم 2 لمحتوى المتغيرة n بالحرف A

- **المطلـوب:**

1. باستخدام التعليمة المتوصل إليها في التمرين 01.06، قدم طريقة الكتابة التي تسمح بإجراء التعديل أعلاه .

✍ التمــريــن 04.06:

لتكن لدينا المتغيرة سلسلة أحرف $y1$ حيث أن:

$$y1 = 'UNIVERSITE'$$

نود إجراء التغييرات التالية:

- استبدال الحرف رقم 1 بالحرف T .
- استبدال الحرف رقم 3 بالحرف G .
- استبدال الحرف رقم 4 بالرمز = .
- استبدال الحرف رقم 5 بالرمز ? .
- استبدال الحرف رقم 6 بالرقم 8 .
- استبدال الحرف رقم 7 بالرمز + .
- استبدال الحرف رقم 8 بالرمز * .
- استبدال الحرف رقم 10 بالحرف Z .

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء التعديلات أعلاه و نشر محتوى بعد إجراء التعديلات.
2. ما هو المحتوى الجديد للمتغيرة $y1$.

✍ التمـريـن 05.06:

لتكن لدينا المتغيرة سلسلة أحرف التالية:

$$ch = 'TIARET'$$

نود إجراء التغييرات التالية:

- استبدال الحرف الأول بالحرف t
- استبدال الحرف الأول بالحرف i
- استبدال الحرف الأول بالحرف a
- استبدال الحرف الأول بالحرف r
- استبدال الحرف الأول بالحرف e
- استبدال الحرف الأول بالحرف t

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء التعديلات أعلاه.
2. ما هو المحتوى الجديد للمتغيرة ch.

✍ التمـريـن 06.06:

لتكن لدينا المتغيرة سلسلة أحرف *chaine* حيث أن:

$$chaine = 'IBNKHALDON'$$

نود إجراء تغييرات أو تعديلات على محتوى المتغيرة *chaine* بحيث يصبح محتواها هو التالي:

$$chaine = '1234567890'$$

- **المطلـوب:**

1. ما هي التغييرات الواجب إجرائها بغرض تغيير محتوى المتغيرة *chaine*.
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء التعديلات المتوصل إليها في المطلوب الأول أعلاه.

✍ التمـريـن 07.06:

لتكن لدينا المتغيرة سلسلة أحرف *wil* حيث أن:

*wil ='WILAYAµDE*TIARET'*

نود إجراء تغييرات أو تعديلات على محتوى المتغيرة *wil* بحيث يصبح محتواها هو التالي:

wil ='WILAYA DE TIARET'

- **المطلـوب:**

1. ما هي التغييرات الواجب إجرائها بغرض تغيير محتوى المتغيرة *wil* .

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء التعديلات المتوصل إليها في المطلوب الأول أعلاه.

✍ التمـريـن 08.06:

لتكن لدينا المتغيرة سلسلة أحرف *NUM* حيث أن:

NUM ='12345098765000090000'

نود إجراء تغييرات أو تعديلات على محتوى المتغيرة *NUM* بحيث يصبح محتواها هو التالي:

NUM ='MODULE INFORMATIQUE'

- **المطلـوب:**

1. ما هي التغييرات الواجب إجرائها بغرض تغيير محتوى المتغيرة *NUM* .

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء التعديلات المتوصل إليها في المطلوب الأول أعلاه.

✍ التمـريـن 09.06:

لتكن لدينا المتغيرة سلسلة أحرف *V* حيث أن:

*V ='**********'*

نود إجراء تغييرات أو تعديلات على محتوى المتغيرة *V* بحيث يصبح محتواها هو التالي:

V ='DIX ETOILE'

- **المطلـوب:**

1. ما هي التغييرات الواجب إجرائها بغرض تغيير محتوى المتغيرة *V*.

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء التعديلات المتوصل إليها في المطلوب الأول أعلاه.

✍ التمـريـن 10.06:

لتكن لدينا المتغيرة سلسلة أحرف *nom* حيث أن:

$nom = 'SETTI * HAMID'$

نود إجراء تغييرات أو تعديلات على محتوى المتغيرة nom بحيث يصبح محتواها هو التالي:

$nom = 'SETTI \quad HAMID'$

- **المطلـوب:**

1. ما هي التغييرات الواجب إجرائها بغرض تغيير محتوى المتغيرة nom.

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء التعديلات المتوصل إليها في المطلوب الأول أعلاه.

✍ التمـريـن 11.06:

1. ما هي الدالة التي تسمح بربط محتوى متغيرتين (معطيتين) أو أكثر من النوع سلسلة أحرف.
2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمـريـن 12.06:

1. ما هي العلامة التي تسمح بربط محتوى متغيرتين (معطيتين) أو أكثر من النوع سلسلة أحرف.
2. قدم طريقة الكتابة للعلامة المتوصل إليها في السؤال الأول.

✍ التمـريـن 13.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$x_1 = 'UNIVERSITE'$

$a = 'DE'$

$b_1 = 'TIARET'$

نريد إجراء عملية الربط و الجمع بين محتوى الثلاث متغيرات x_1 و a و b_1

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام علامة الجمع + .

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام الدالة **concat( )** .

3. وضح على الشاشة نتيجة عملية الربط .

✍ التمــريــن 14.06:

لتكن متغيرات التمرين 14.06 أعلاه و لتكن المتغيرة من النوع سلسلة أحرف التالية:

esp_1:=' '

نريد إجراء عملية الربط و الجمع بين محتوى المتغيرات التالية:

x_1 **et** *esp_1* **et** *a* **et** *esp_1* **et** *b_1*

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام علامة الجمع + .
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام الدالة **concat( )** .
3. وضح على الشاشة نتيجة عملية الربط .

✍ التمــريــن 15.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$ch1 = 'LA'$

$ch2 = 'VILLE'$

$ch3 = 'DE'$

$ch4 = 'TIARET'$

$ch5 = 'EST'$

$ch6 = 'BELLE'$

$esp = ' '$

نريد الربط بين محتوى المتغيرات التالية:

- *ch*1 و *ch*6 و تخزين النتيجة في المتغيرة *ch*16
- *ch*6 و *ch*1 و تخزين النتيجة في المتغيرة *ch*61
- *ch*6 و *ch*6 و تخزين النتيجة في المتغيرة *ch*66
- *ch*2 و *ch*4 و *ch*2 تخزين النتيجة في المتغيرة *ch*242
- *ch*5 و *ch*2 و تخزين النتيجة المتغيرة *ch*52
- *ch*1 و *ch*2 و *ch*6 و تخزين النتيجة في المتغيرة *ch*126
- *ch*6 و *ch*2 و *ch*5 تخزين النتيجة في المتغيرة *ch*625

- *ch*4 و *ch*1 و *ch*6 و *ch*3 و تخزين النتيجة في المتغيرة *ch*4163
- *ch*6 و *ch*5 و *ch*4 و *ch*3 و *ch*2 و *ch*1 تخزين النتيجة في المتغيرة *ch*
- *ch*1 و *ch*2 و *ch*3 و *ch*4 و *ch*5 و *ch*6 تخزين النتيجة في المتغيرة *chh*
- *ch*1 و *esp* و *ch*2 و *esp* و *ch*3 و *esp* و *ch*4 و *esp* و *ch*5 و *esp* و *ch*6 تخزين النتيجة في المتغيرة *chhh*

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط أعلاه باستخدام علامة الجمع + .

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط أعلاه باستخدام الدالة **( )concat** .

3. وضح على الشاشة نتائج عمليات الربط .

✍ التمـريـن 16.06:

لتكن لدينا المعطيات من النوع سلسلة أحرف التالية:

FACULTE

DES

SCIENCES

ECONOMIQUES

نريد ربط هذه المعطيات و تخزين نتيجة عملية الربط في متغيرة من النوع سلسلة أحرف *k*

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام علامة الجمع + .

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام الدالة **( )concat** .

3. وضح على الشاشة نتيجة عملية الربط .

✍ التمـريـن 17.06:

حتى يكون محتوى المتغيرة *k* للتمرين 16.06 له معنى و مقروء نقوم بإدراج الفراغات التالية:

- فراغ واحد بين *FACULTE* و *DES*

- فراغ واحد بين *DES* و *SCIENCES*
- فراغ واحد بين *SCIENCES* و *ECONOMIQUES*

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام علامة الجمع + .

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام الدالة **concat( )** .

3. وضح على الشاشة نتيجة عملية الربط .

✍ التمـريـن 18.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

IBN KHALDOUN

UNIVERSITE

$ch1 = 'LA'$

$ch2 = 'WILAYA'$

$ch3 = 'DE'$

TIARET

$ch4 = ' '$

نريد الربط بين ما يلي:

- *ch*1 و *ch*2 و *ch*3 و *TIARET* و تخزين النتيجة في المتغيرة *ch*123 .
- *ch*1 و *ch*4 و *ch*2 و *ch*4 و *ch*3 و *ch*4 و *TIARET* و تخزين النتيجة في المتغيرة *ch*124 .
- *UNIVERSITE* و *ch*3 و *ch*1 و *ch*2 و *ch*3 و *TIARET* و تخزين النتيجة في المتغيرة *ch*125 .
- *UNIVERSITE* و *ch*4 و *ch*3 و *ch*4 و *ch*1 و *ch*4 و *ch*2 و *ch*4 و *ch*3 و *ch*4 و *TIARET* و تخزين النتيجة في المتغيرة *ch*126 .

✍ التمـريـن 19.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

IBN KHALDOUN

UNIVERSITE

$ch1 = 'LA'$

$ch2 = 'WILAYA'$

$ch3 = 'DE'$

TIARET

$ch4 = ' '$

نريد الربط بين ما يلي:

- $ch1$ و $ch2$ و $ch3$ و *TIARET* و تخزين النتيجة في المتغيرة $ch123$.
- $ch1$ و $ch4$ و $ch2$ و $ch4$ و $ch3$ و $ch4$ و $TIARE'$ و تخزين النتيجة في المتغيرة $ch124$.
- *UNIVERSITE* و $ch3$ و $ch1$ و $ch2$ و $ch3$ و *TIARET* و تخزين النتيجة في المتغيرة $ch125$.
- *UNIVERSITE* و $ch4$ و $ch3$ و $ch4$ و $ch1$ و $ch4$ و $ch2$ و $ch4$ و $ch3$ و $ch4$ و *TIARET* و تخزين النتيجة في المتغيرة $ch126$.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط أعلاه باستخدام علامة الجمع + .
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط أعلاه باستخدام الدالة **concat( )** .
3. وضح على الشاشة نتائج عمليات الربط .

✍ التمـريـن 20.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$a1 = JE$

$a2 = TE$

$a3 = SOUHAITE$

$a4 = UNE$

$a5 = BONNE$

$a6 = NUIT$

$a7 = ' '$

نريد الربط بين محتوى المتغيرات التالية:

- $a4$ و $a6$ و تخزين النتيجة في المتغيرة $a46$
- $a4$ و $a7$ و $a6$ و تخزين النتيجة في المتغيرة $a476$
- $a6$ و $a6$ و تخزين النتيجة في المتغيرة $a66$
- $a6$ و $a7$ و $a6$ و تخزين النتيجة في المتغيرة $a676$
- $a1$ و $a2$ و $a3$ و تخزين النتيجة في المتغيرة $a123$
- $a1$ و $a7$ و $a2$ و $a7$ و $a3$ و تخزين النتيجة في المتغيرة $a17273$
- $a1$ و $a2$ و تخزين النتيجة في المتغيرة $a12$
- $a1$ و $a7$ و $a2$ و تخزين النتيجة في المتغيرة $a172$
- $a5$ و $a6$ و تخزين النتيجة في المتغيرة $a56$
- $a5$ و $a7$ و $a6$ و تخزين النتيجة في المتغيرة $a576$
- $a1$ و $a2$ و $a3$ و $a4$ و $a5$ و $a6$ و تخزين النتيجة في المتغيرة $a123456$
- $a1$ و $a7$ و $a2$ و $a7$ و $a3$ و $a7$ و $a4$ و $a7$ و $a5$ و $a7$ و $a6$ و تخزين النتيجة في المتغيرة a

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط أعلاه باستخدام علامة الجمع + .
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط أعلاه باستخدام الدالة **concat( )** .
3. وضح على الشاشة نتائج عمليات الربط .

✍ التمـريـن 20.06:

لدينا المعطيات من النوع سلسلة أحرف التالية:

2011

/

2012

ANNEE

UNIVERSITAIRE

نريد ربط هذه المعطيات و تخزين نتيجة عملية الربط في متغيرة من النوع سلسلة أحرف k

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام علامة الجمع + .

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عملية الربط أعلاه باستخدام الدالة **concat( )** .

3. وضح على الشاشة نتائج عمليات الربط .

✍ التمـريـن 21.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

$ch1 = PROGRAMMTION$

LANGAGE

$ch2 = EN$

$ch3 = PASCAL$

$ch5 = 'DE'$

LIVRE

$ch4 = ' '$

نريد الربط بين ما يلي:

- *LANGAGE* و $ch3$ و تخزين النتيجة في المتغيرة $ch03$.
- *LANGAGE* و $ch4$ و $ch3$ و تخزين النتيجة في المتغيرة $ch043$.
- *LIVRE* و *LANGAGE* و $ch3$ و تخزين النتيجة في المتغيرة $ch003$.
- *LIVRE* و $ch4$ و *LANGAGE* و $ch4$ و $ch3$ و تخزين النتيجة في المتغيرة $ch04043$.
- *LIVRE* و $ch5$ و $ch1$ و $ch2$ و *LANGAGE* و $ch3$ و تخزين النتيجة في المتغيرة ch.
- *LIVRE* و $ch4$ و $ch5$ و $ch4$ و $ch1$ و $ch4$ و $ch2$ و $ch4$ و *LANGAGE* و $ch4$ و $ch3$ و تخزين النتيجة في المتغيرة chh.

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط أعلاه باستخدام علامة الجمع + .
2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط أعلاه باستخدام الدالة **concat( )** .
3. وضح على الشاشة نتائج عمليات الربط .

✍ التمـريـن 22.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

$a =$ ' '

$b = MATHEMATIQUE$

$c = MODULE$

$d = DU$

DE

$LIVRE$

$e = NOTE$

نريد الحصول على الشاشة على ما يلي:

النشر على الشاشة

MODULEDEMATHEMATIQUE
MODULE DE MATHEMATIQUE
LIVREDEMATHEMATIQUE
LIVRE DE MATHEMATIQUE
NOTE DU MODULE DE MATHEMATIQUE
NOTE LIVRE MODULEDEMATHEMATIQUE

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بإجراء عمليات الربط (باستخدام العلامة + ثم الدالة **concat( )**) بحيث عند نشر نتائج عملية الربط يتم الحصول على ما هو موضح على الشاشة أعلاه.

✍ التمـريـن 23.06:

1. ما هي الدالة التي تسمح بمعرفة طول متغيرة أو معطية من النوع سلسلة أحرف.

2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمـريـن 24.06:

لتكن لدينا المتغيرات من النوع سلسلة أحرف التالية:

$x = DATEDENAISSANCE - DE * MOHAMED$

$y = LA \;\; WILAYA \;\; DE \;\; TIARET$

$z = LA \;\; VILLE \;\; DE \;\; TIARET \;\; TRES \;\; BELLE$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب طول محتوى المتغيرات أعلاه.

✍ التمـريـن 24.06:

لتكن لدينا المعطيات من النوع سلسلة أحرف التالية:

$MODULE \;\; INFORMATIQUE$

$ANNEE \;\; UNIVERSITAIRE \;\; 2011/2012$

$BON03\mu * £v \;\; COURAGEKLM\%£432$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب طول المعطيات أعلاه.

✍ التمـريـن 25.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

$proGRAMMATION \;\; EN \;\; PAscal$

$a = 2 \;\; EME \;\; ANNEE \;\; SCIENCES \;\; COMMERCIALLES$

$INStructions$

$b = 46587093 + =,:!\grave{u} *$

46587093+=,:!ù

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب طول المعطيات و محتوى المتغيرات أعلاه.

✍ التمــريــن 26.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

DE LA PART DE HAMID

$x_1 = 'DIRECTION\ \ DE\ \ LA\ \ FORMATION'$

$x_2 = 'mohamed'$

MicroORDINATEUR

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بحساب طول المعطيات و محتوى المتغيرات أعلاه.

✍ التمــريــن 27.06:

1. ما هي الدالة التي تسمح باستخلاص و استنتاج مجموعة جزئية من الحروف من محتوى متغيرة أو معطية من النوع سلسلة أحرف.

2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمــريــن 28.06:

لتكن x متغيرة من النوع سلسلة أحرف و التي تساوي ما يلي:

$$x = LIVREDEPROGRAMMATIONENLANGAGEPASCAL$$

لنفرض أننا نريد استخلاص مجموعات جزئية من الحروف انطلاقا من محتوى المتغيرة x الذي هو: $LIVREDEPROGRAMMATIONENLANGAGEPASCAL$. من بين المجموعات الجزئية الممكن استخلاصها نذكر ما يلي:

- المجموعة الجزئية $LIVRE$: تتكون من $j = 5$ حروف أو أحرف إبتداءا من الحرف الأول $i = 1$ لمحتوى المتغيرة x.
- المجموعة الجزئية $LANGAGEPASCAL$: تتكون من $j = 13$ حروف أو أحرف إبتداءا من الحرف الثالث و العشرون $i = 23$ لمحتوى المتغيرة x.
- المجموعة الجزئية $PASCAL$: تتكون من $j = 6$ حروف أو أحرف إبتداءا من الحرف الثلاثون $i = 30$ لمحتوى المتغيرة x.
- المجموعة الجزئية $LANGAGE$: تتكون من $j = 7$ حروف أو أحرف إبتداءا من الحرف الثالث و العشرون $i = 23$ لمحتوى المتغيرة x.

- المجموعة الجزئية *PROGRAMMATIONEN* : تتكون من $j=15$ حروف أو أحرف إبتداءا من الحرف الثامن $i=8$ لمحتوى المتغيرة x.
- المجموعة الجزئية *DE* : تتكون من $j=2$ حروف أو أحرف إبتداءا من الحرف السادس $i=6$ لمحتوى المتغيرة x.
- المجموعة الجزئية *EN* : تتكون من $j=2$ حروف أو أحرف إبتداءا من الحرف السادس $i=6$ لمحتوى المتغيرة x.
- المجموعة الجزئية *T* : تتكون من $j=1$ حروف أو أحرف إبتداءا من الحرف السابع عشر $i=17$ لمحتوى المتغيرة x.
- المجموعة الجزئية *REDEPROGRAMMATIONENLANGAGEPASCA* : تتكون من $j=31$ حروف أو أحرف إبتداءا من الحرف الرابع $i=4$ لمحتوى المتغيرة x.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الاستخلاص أعلاه.

✍ التمـريـن 29.06:

لتكن لدينا المتغيرة من النوع سلسلة أحرف التالية:

$$ch = LAWILAYADETIARET$$

نريد إجراء مجموعة من عمليات الاستخلاص من محتوى المتغيرة *ch* كما يلي:

- استخلاص $j=6$ أحرف ابتداءا من الحرف رقم $i=3$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch63$.
- استخلاص $j=2$ أحرف ابتداءا من الحرف رقم $i=9$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch29$.
- استخلاص $j=4$ أحرف ابتداءا من الحرف رقم $i=1$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch41$.
- استخلاص $j=7$ أحرف ابتداءا من الحرف رقم $i=2$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch72$.
- استخلاص $j=16$ أحرف ابتداءا من الحرف رقم $i=1$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch161$.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الاستخلاص أعلاه.

✍ التمــريــن 30.06:

لتكن لدينا المعطية النوع سلسلة أحرف التالية:

$$FARIDA12304BONNE4500$$

نريد إجراء مجموعة من عمليات الاستخلاص على المعطية أعلاه كما يلي:

- استخلاص $j=6$ أحرف ابتداءا من الحرف رقم $i=1$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch61$.
- استخلاص $j=5$ أحرف ابتداءا من الحرف رقم $i=7$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch57$.
- استخلاص $j=9$ أحرف ابتداءا من الحرف رقم $i=12$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch912$.
- استخلاص $j=4$ أحرف ابتداءا من الحرف رقم $i=17$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch417$.
- استخلاص $j=3$ أحرف ابتداءا من الحرف رقم $i=12$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف $ch312$.
- استخلاص $j=20$ أحرف ابتداءا من الحرف رقم $i=1$ و تخزين نتيجة عملية الاستخلاص في المتغيرة من النوع سلسلة أحرف chh.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الاستخلاص أعلاه.

✍ التمــريــن 31.06:

لتكن لدينا المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

$$MIcRO\ \ ORDInatEUR$$

$$a=1\ \ ERE\ \ AnnEE\ \ masTER$$

نريد إجراء مجموعة من عمليات الاستخلاص على المعطيات أعلاه كما يلي:

- استخلاص $j=6$ أحرف ابتداءا من الحرف رقم $i=3$ من المعطية $MIcRO\ \ ORDInatEUR$
- استخلاص و نشر $j=10$ أحرف ابتداءا من الحرف رقم $i=7$ من المعطية $MIcRO\ \ ORDInatEUR$
- استخلاص و نشر $j=10$ أحرف ابتداءا من الحرف رقم $i=1$ من محتوى المتغيرة a
- استخلاص و نشر $j=8$ أحرف ابتداءا من الحرف رقم $i=7$ من محتوى المتغيرة a

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الاستخلاص أعلاه.

✍ التمــريــن 32.06:

لتكن المتغيرة من النوع سلسلة أحرف التالية:

$$G = le \quad MONDE \quad et \quad petit \quad 1234909$$

انطلاقا من محتوى المتغيرة G، نريد استخلاص المعطيات التالية:

- 123
- *le MONDE*
- *E et petit* 1
- 1234909
- *le MONDE et petit*

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الاستخلاص أعلاه.

✍ التمــريــن 33.06:

لتكن المعطية من النوع سلسلة أحرف التالية:

ANNEE UNIVERSITAIRE 2012/2013

انطلاقا من المعطية أعلاه، نريد استخلاص المعطيات التالية:

- 2012
- /2013
- *ANNEE UNIVERSITAIRE* 2012
- *UNIVERSITAIRE* 2012/2013
- *ANNEE UNIVERSITAIRE*
- 2012/2013
- *AUNIVERSITAIRE* 2012/2

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الاستخلاص أعلاه.

✍ التمــريــن 34.06:

لتكن المتغيرة و المعطية من النوع سلسلة أحرف التاليتين:

$$x_1 = CENTRE \quad COMMERCIAL \quad CARFOUR$$

$$LIVRE \quad DE \quad PROGRAMMATION \quad LINEAIRE$$

انطلاقا من المتغيرة و المعطية، نريد استخلاص المعطيات التالية:

- *CENTRE*
- *COMMERCIAL CARFOUR*
- *RE COMMERCIAL CAR*
- *LIVRE D*
- *PROGRAMMATION LINEAIRE*
- *LIVRE DE PR*

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الاستخلاص أعلاه.

✍ التمــريــن 35.06:

لتكن المتغيرات و المعطيات من النوع سلسلة أحرف التالية:

$$y_1 = 1234567890$$

$$y_2 = ?/\S\%\text{¨£¨\$*!} = +$$

نريد إجراء مجموعة من عمليات الاستخلاص على المعطيات أعلاه كما يلي:

- استخلاص $j = 6$ أحرف ابتداءا من الحرف رقم $i = 3$
- استخلاص $j = 5$ أحرف ابتداءا من الحرف رقم $i = 4$
- استخلاص $j = 1$ أحرف ابتداءا من الحرف رقم $i = 9$
- استخلاص $j = 9$ أحرف ابتداءا من الحرف رقم $i = 1$
- استخلاص $j = 3$ أحرف ابتداءا من الحرف رقم $i = 3$
- استخلاص $j = 7$ أحرف ابتداءا من الحرف رقم $i = 3$

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الاستخلاص أعلاه.

✍ التمــريــن 36.06:

1. ما هي الدالة التي تسمح بحذف مجموعة جزئية من الحروف من محتوى متغيرة أو معطية من النوع سلسلة أحرف.
2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمــريــن 37.06:

لتكن x متغيرة من النوع سلسلة أحرف و التي تساوي ما يلي:

$$y = LIVRE55?DE098LANGAGE + \%\mu PASCAL$$

نريد حذف مجموعات جزئية من الحروف انطلاقا من محتوى المتغيرة y من بين المجموعات الجزئية الممكن حذفها نذكر ما يلي:

- المجموعة الجزئية $55?$: تتكون من $j = 3$ حروف أو أحرف إبتداءا من الحرف الأول $i = 6$ لمحتوى المتغيرة x.
- المجموعة الجزئية $\%\mu PASCAL+$: تتكون من $j = 9$ حروف أو أحرف إبتداءا من الحرف الأول $i = 21$ لمحتوى المتغيرة x.
- المجموعة الجزئية 098: تتكون من $j = 3$ حروف أو أحرف إبتداءا من الحرف الأول $i = 11$ لمحتوى المتغيرة x.

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الحذف أعلاه.

2. نشر المحتوى الجديد للمتغيرة y.

✍ التمــريــن 38.06:

لتكن لدينا المتغيرة من النوع سلسلة أحرف التالية:

$$ch = MODULE67890INFORMATIQUE\mu *+=?\&\S$$

نريد إجراء على محتوى المتغيرة ch عمليات الحذف التالية:

- حذف $j = 5$ أحرف ابتداءا من الحرف رقم $i = 7$.
- حذف $j = 7$ أحرف ابتداءا من الحرف رقم $i = 19$
- حذف $j = 1$ أحرف ابتداءا من الحرف رقم $i = 1$
- حذف $j = 4$ أحرف ابتداءا من الحرف رقم $i = 10$

- حذف $14 = j$ أحرف ابتداءا من الحرف رقم $3 = i$

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الحذف أعلاه.

2. نشر المحتوى الجديد للمتغيرة ch.

✍ التمـريـن 39.06:

لتكن المتغيرة من النوع سلسلة أحرف التالية:

$$f = 27\mu\mu AB17890MD\mu DFRU$$

نريد إجراء على محتوى المتغيرة f عمليات الحذف التالية:

- حذف $4 = j$ أحرف إبتداءا من الحرف رقم $3 = i$.
- حذف $7 = j$ أحرف إبتداءا من الحرف رقم $7 = i$.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الحذف أعلاه.

2. نشر المحتوى الجديد للمتغيرة f .

✍ التمـريـن 39.06:

لتكن المتغيرة من النوع سلسلة أحرف التالية:

$$g = plan\S60\grave{a}\S\S NATIONALE$$

نريد إجراء على محتوى المتغيرة g عمليات الحذف التالية:

- حذف $6 = j$ أحرف إبتداءا من الحرف رقم $5 = i$.
- حذف $1 = j$ أحرف إبتداءا من الحرف رقم $13 = i$.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الحذف أعلاه.

2. نشر المحتوى الجديد للمتغيرة g.

✍ التمـريـن 40.06:

لتكن المتغيرات من النوع سلسلة أحرف التالية:

$$g = AZERTY12345489JGH$$

$$H = HGDOPRU8764\S/.?\%$$

$$Y = 1234567890AZERTYIOP$$

$$S = ?./\S\%\mu£¨PRU8764\S/.?\%$$

$$T = DFS./\S\%\mu£BNV8790.?\%$$

نريد إجراء على محتوى المتغيرات أعلاه عمليات الحذف التالية:

- حذف $j = 3$ أحرف إبتداءا من الحرف رقم $i = 5$.
- حذف $j = 2$ أحرف إبتداءا من الحرف رقم $i = 3$.
- حذف $j = 1$ أحرف إبتداءا من الحرف رقم $i = 1$.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الحذف أعلاه.

2. نشر المحتوى الجديد للمتغيرات.

✍ التمـريـن 41.06:

1. ما هي الدالة التي تسمح **دمـج** (Insertion) محتوى متغيرة من النوع سلسلة أحرف في أو ضمن محتوى متغيرة أخرى من النوع سلسلة أحرف.

2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمـريـن 42.06:

لتكن المتغيرات من النوع سلسلة أحرف التالية:

$$ch1 = MOISDEJANVIER$$

$$ch2 = DE$$

$$ch4 = ANNEE2012$$

$$ch3 = ' \ '$$

نريد إجراء مجموعة من عمليات الدمج أو الإدماج على المتغيرات أعلاه كما يلي:

- دمج محتوى المتغيرة $ch2$ في أو ضمن محتوى المتغيرة $ch1$ ابتداءا من الحرف رقم $j = 5$.
- دمج محتوى المتغيرة $ch3$ في ضمن المحتوى الجديد للمتغيرة $ch1$ ابتداءا من الحرف رقم $j = 5$.
- دمج محتوى المتغيرة $ch3$ في أو ضمن المحتوى الجديد للمتغيرة $ch1$ ابتداءا من الحرف رقم $j = 8$.

- دمج محتوى المتغيرة $ch4$ في ضمن المحتوى الجديد للمتغيرة $ch2$ ابتداءا من الحرف رقم $j = 2$.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الدمج أعلاه.

2. نشر المحتوى الجديد لجميع المتغيرات.

✍ التمـريـن 43.06:

لتكن المتغيرات من النوع سلسلة أحرف التالية:

$$a = '12345'$$

$$b = 'abc'$$

$$c = '\%/£\$'$$

$$d = ' \ '$$

نريد إجراء عمليات الدمج على المتغيرات أعلاه كما يلي:

1. دمج محتوى المتغيرة d ضمن محتوى المتغيرة a ابتداءا من الحرف رقم $j = 4$.

2. دمج محتوى المتغيرة d ضمن محتوى المتغيرة c ابتداءا من الحرف رقم $j = 3$.

3. دمج محتوى المتغيرة a ضمن محتوى المتغيرة c ابتداءا من الحرف رقم $j = 2$.

4. دمج محتوى المتغيرة b ضمن محتوى المتغيرة c ابتداءا من الحرف رقم $j = 3$.

5. دمج محتوى المتغيرة b ضمن محتوى المتغيرة a ابتداءا من الحرف رقم $j = 2$.

– **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الدمج أعلاه.

2. نشر المحتوى الجديد لجميع المتغيرات.

✍ التمـريـن 44.06:

لتكن المتغيرات من النوع سلسلة أحرف التالية:

$$w = ' \ '$$

$$x = 'delapart'$$

$$y = 'de'$$

$$z = 'toi'$$

نريد إجراء عمليات الدمج على المتغيرات أعلاه كما يلي:

1. دمج محتوى المتغيرة w ضمن محتوى المتغيرة x ابتداءا من الحرف رقم $j=3$.
2. دمج محتوى المتغيرة w ضمن محتوى المتغيرة x ابتداءا من الحرف رقم $j=6$.
3. دمج محتوى المتغيرة w ضمن محتوى المتغيرة x ابتداءا من الحرف رقم $j=11$.

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز عمليات الدمج أعلاه.
2. نشر المحتوى الجديد لجميع المتغيرات.

✍ التمـريـن 45.06:

1. ما هي الدالة التي تسمح بتقديم شفرة (Le Code ASCII) أي حرف أو رمز أو عدد .
2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمـريـن 46.06:

لتكن الحروف و الحروف الخاصة و الأعداد التالية:

$$B, C, 2, 3, /, :, !, \S, m, n, <, >$$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم شفرة (Le Code ASCII) المعطيات من النوع أحرف أعلاه.

✍ التمـريـن 47.06:

لتكن الحروف و الحروف الخاصة و الأعداد التالية:

$$?, \ddot{}\$, 6, 7, F, f$$

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم شفرة (Le Code ASCII) المعطيات من النوع أحرف أعلاه.

✍ التمـريـن 48.06:

لتكن لدينا المتغيرات و المعطيات من النوع أحرف التالية:

$$a = 'Y'$$

$$b = 'Z'$$

$$W$$

X

Y

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم شفرة (Le Code ASCII) محتوى المتغيرات و المعطيات أعلاه.

✍ التمـريـن 49.06:

لتكن المعطيات من النوع أحرف التالية:

a , b , A , B , i , I , 3 , * , w , W

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم شفرة (Le Code ASCII) المعطيات أعلاه.

✍ التمـريـن 50.06:

لتكن المعطيات من النوع أحرف التالية:

" ,] , [, = , } , {

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم شفرة (Le Code ASCII) المعطيات أعلاه.

✍ التمـريـن 51.06:

1. ما هي الدالة التي تسمح بتقديم الحرف الخاص الموافق لأي شفرة (Le Code ASCII).

2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمـريـن 52.06:

لنفرض أننا نريد معرفة الحروف الخاصة الموافقة للشفرات (Les Codes ASCII) التالية:

65 , 66 , 70 , 250 , 251 , 253

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحروف الخاصة الموافقة للشفرات أعلاه.

✍ التمــريــن 53.06:

لنفرض أننا نريد معرفة الحروف الخاصة الموافقة للشفرات (Les Codes ASCII) التالية:

71 , 72 , 73 , 80 , 85 , 88

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحروف الخاصة الموافقة للشفرات أعلاه.

✍ التمــريــن 54.06:

لنفرض أننا نريد معرفة الحروف الخاصة الموافقة للشفرات (Les Codes ASCII) التالية:

150 , 151 , 154 , 225 , 250 , 88

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحروف الخاصة الموافقة للشفرات أعلاه.

✍ التمــريــن 55.06:

لتكن الشفرات (Les Codes ASCII) التالية:

90 , 91 , 93 , 100 , 175 , 176

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحروف الخاصة الموافقة للشفرات أعلاه.

✍ التمــريــن 56.06:

1. ما هي الدالة التي تسمح تسمح بتقديم الحرف الخاص الذي **يلي** حرف خاص معين.

2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمــريــن 57.06:

لنفرض أننا نريد معرفة الحرف الخاص الذي يلي كل حرف خاص من الأحرف الخاصة التالية:

A , X , t , x , ? , =

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يلي كل حرف خاص من الأحرف الخاصة أعلاه.

✍ التمـريـن 58.06:

لتكن الأحرف الخاصة التالية:

1 , + , H , h , J , %

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يلي كل حرف خاص من الأحرف الخاصة أعلاه.

✍ التمـريـن 59.06:

لتكن الأحرف الخاصة التالية:

3 , 5 , < , B , > , O , 123 , 200

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يلي كل حرف خاص من الأحرف الخاصة أعلاه.

✍ التمـريـن 60.06:

لتكن الأحرف الخاصة التالية:

a , b , c , d , E , e , F , f

- **المطلـوب:**

2. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يلي كل حرف خاص من الأحرف الخاصة أعلاه.

✍ التمـريـن 61.06:

1. ما هي الدالة التي تسمح تسمح بتقديم الحرف الخاص الذي **يسبق** حرف خاص معين.
2. قدم طريقة الكتابة للدالة المتوصل إليها في السؤال الأول.

✍ التمـريـن 62.06:

لنفرض أننا نريد معرفة الحرف الخاص الذي يلي كل حرف خاص من الأحرف الخاصة التالية:

A , X , t , x , ? , =

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يسبق كل حرف خاص من الأحرف الخاصة أعلاه.

✍ التمـريـن 63.06:

لتكن الأحرف الخاصة التالية:

1 , + , H , h , J , %

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يسبق كل حرف خاص من الأحرف الخاصة أعلاه.

✍ التمـريـن 64.06:

لتكن الأحرف الخاصة التالية:

3 , 5 , < , B , > , O , 123 , 200

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يسبق كل حرف خاص من الأحرف الخاصة أعلاه.

✍ التمـريـن 65.06:

لتكن الأحرف الخاصة التالية:

a , b , c , d , E , e , F , f

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يسبق كل حرف خاص من الأحرف الخاصة أعلاه.

✍ التمـريـن 66.06:

لنفرض أننا نريد معرفة الأحرف الخاصة التي تلي الأحرف الخاصة التالية:

B , Z , # , I , + , ?

- **المطلـوب:**

1. أكتب البرنامج بلغة الـ PASCAL الذي يسمح بتقديم الحرف الخاص الذي يسبق كل حرف خاص من الأحرف الخاصة أعلاه.

الفصــل 📖 السابــع

نشـــر (طبـاعــة) المعطيـــات

Affichage (Edition) Des Données

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- مختلف التعليمات المستخدمة لنشر و نسخ محتوى كل أنواع المتغيرات.
  1. التعليمة *write* .
  2. التعليمة *writeln* .

- مختلف طرق كتابة التعليمات المستخدمة لنشر و نسخ محتوى المتغيرات.
  1. مختلف طرق كتابة التعليمة *write* .

Les Différentes Syntaxe De L'instruction write

  2. مختلف طرق كتابة التعليمة *writeln* .

Les Différentes Syntaxe De L'instruction writeln

  3. مختلف أشكال النشر

Les Différentes Formats d'Affichage

  4. الفرق بين التعليمتين *Write* و *Writeln*

La Différences Etre Les Instructions *Write* et *Writeln*

  5. استخدامات أخرى للتعليمتين *Write* و *Writeln*

Autres Utilisations Des Instructions *Write* et *Writeln*

01.07 – تمهيـد:

بعد المرحلة الثالثة الممثلة في معالجة المعطيات التي تم إدخالها في الخطوة الثانية، يأتي دور الخطوة و المرحلة الرابعة الممثلة في إظهار و نشر (نسخ) على الشاشة (Affichage sur écran) النتائج المتحصل عليها من عملية المعالجة. التعليمات (Les Instructions) التي تسمح بنشر و نسخ النتائج على الشاشة هي:

1. L'instruction ***write*** التعليمة
2. L'instruction ***writeln*** التعليمة

سوف يتم التطرق بالتفضيل إلى كل تعليمة من التعليمات المذكورة أعلاه.

02.07 – التعليمـة L'instruction Write

التعليمة ***write*** تسمح بنشر و نسخ و إظهار على الشاشة (Affichage sur écran) محتوى متغيرة ما. طريقة كتابة التعليمة La Syntaxe De L'instruction ***write*** هي التالية:

Write **(** *Variable* **) ;**

حيث أن:

- ***Write*** : تمثل التعليمة التي تسمح بنشر و نسخ و إظهار على الشاشة محتوى متغيرة ما.
- **( )** : عبارة عن قوسين.
- *Variable*: تمثل المتغيرة المراد أو التي نريد نشر محتواها. هذه المتغيرة قد تكون من النوع الصحيح أو الحقيقي أو من النوع حرف أو من النوع سلسلة أحرف.
- **;** : تمثل فاصلة منقوطة Point Virgule

من خلال المثال 01.07 أدناه نقوم من خلاله بنشر محتوى المتغيرة a.

مثـــال 01.07:

Write (*a* ***) ;***

و بغرض التوضيح أكثر حول كيفية استخدام التعليمة ***Write*** ضمن البرنامج بلغة الـ PASCAL نأخذ المثال التالي:

مثـــال 02.07:

ليكن البرنامج الذي من خلاله نقوم بنشر و إظهار على الشاشة (Sur Ecran) محتوى المتغيرة الصحيحة *som* التي تمثل مجموع المتغيرتين الصحيحتين x و y اللتين تم إدخال قيمتيهما عن طريق

البرنامج أي عن طريق التوجيه المباشر.

```
program Affichage_Sur_Ecran ;
var  x , y , som : integer;
begin
        x := 1210 ;
        y := 4500 ;
        som := x+y ;
        write (  som ) ;
        readln ;
end.
```

السطر

```
write (som) ;
```

يسمح بنشر على الشاشة (Sur Ecran) محتوى المتغيرة الصحيحة *som* كما يلي:

النشر على الشاشة

```
5710_
```

الكمبيوتر أو الحاسوب يقوم بنشر محتوى المتغيرة *som* إبتداءا من السطر الأول (Première Ligne) و العمود الأول (Première Colonne) للشاشة. بما أن محتوى المتغيرة *som* هو 5710 و الذي يتكون من أربعة (4) أرقام فإن عدد الأعمدة التي سوف يستخدمها الكمبيوتر لنشر محتوى المتغيرة *SOM* يساوي أربعة أعمدة حيث يقوم الحاسوب بنشر:

- الرقم 5 في العمود الأول.
- الرقم 7 في العمود الثاني.
- الرقم 1 في العمود الثالث.
- الرقم 0 في العمود الرابع.

و بعدما ينتهي الكمبيوتر من نشر محتوى المتغيرة *som* الذي هو 5710 يتموضع (_ Le Curseur) عند أو في العمود الخامس كما هو موضح على الشاشة أعلاه.

حيث أن:

- _ يرمز على (Symbolise Le Curseur)

في بعض المرات لا نرغب في نشر محتوى متغيرة واحدة فقط و إنما نرغب في نشر محتوى أكثر من متغيرة واحدة أي متغيرتين أو أكثر مثلما يوضحه برنامج المثال 03.07 أدناه:

مثـــال 03.07:

```
program Affichage_Sur_Ecran ;
var  x , y , som , soust , mult : integer;
     nom , prenom : string ;
begin
     x := 1210 ;
     y := 4500 ;
     nom :=' MAKHLOUF' ;
     prenom :='AEK' ;
     som := x+y ;
     soust := y-x ;
     mult := x*y ;
     write (som) ;
     write (soust) ;
     write (mult) ;
     write (nom) ;
     write (prenom) ;
     readln ;
end.
```

البرنامج أعلاه يقوم بنشر ما يلي:

- محتوى المتغيرة *som* الذي يساوي 5710 و هذا إبتداءا من العمود الأول للشاشة إلى غاية العمود الرابع و بعدها يتموضع (_ Le Curseur) في العمود الخامس للشاشة ليبدأ في نشر محتوى المتغيرة الموالية أي محتوى المتغيرة *soust*.

- محتوى المتغيرة *soust* الذي يساوي 3290 و هذا إبتداءا من العمود الخامس للشاشة إلى غاية العمود الثامن و بعدها يتموضع (_ Le Curseur) في العمود التاسع للشاشة ليبدأ في نشر محتوى المتغيرة الموالية *mult*.
- محتوى المتغيرة *mult* الذي يساوي 5445000 إبتداءا من العمود التاسع للشاشة إلى غاية العمود الخامس عشر و بعدها يتموضع (_ Le Curseur) في العمود السادس عشر للشاشة ليبدأ في نشر محتوى المتغيرة الموالية *nom*.
- محتوى المتغيرة *nom* الذي يساوي أو الذي هو MAKHLOUF إبتداءا من العمود السادس عشر للشاشة إلى غاية العمود الثالث و العشرون و بعدها يتموضع (_ Le Curseur) في العمود الرابع و العشرون للشاشة ليبدأ في نشر محتوى المتغيرة الموالية *prenom*.
- محتوى المتغيرة *prenom* الذي يساوي أو الذي هو AEK إبتداءا من العمود الرابع و العشرون للشاشة إلى غاية العمود السادس و العشرون و بعدها يتموضع (_ Le Curseur) في العمود السابع و العشرون للشاشة كما هو موضح على الشاشة أدناه:

النشر على الشاشة

```
571032905445000MAKHLOUFAEK_
```

ملاحظـة 01.07 :

التعليمات الخمسة المتعلقة بنشر محتوى المتغيرات *som* ، *soust* ، *mult* ، *nom* ، *prenom*

```
write ( som ) ;
write ( soust ) ;
write ( mult ) ;
write ( nom ) ;
write ( prenom ) ;
```

يمكن تعويضها بتعليمة واحدة فقط كما يلي:

```
write ( som , soust , mult , nom , prenom ) ;
```

و عليه يصبح برنامج المثال 03.07 أعلاه كما يلي:

مثـــال 04.07:

```
program Affichage_Sur_Ecran ;
var  x , y , som , soust , mult : integer ;
        nom , prenom : string ;

begin
    x:= 1210 ;
    y:= 4500 ;
    nom :=' MAKHLOUF' ;
    prenom :='AEK' ;
    som:= x+y ;
    soust:= y-x ;
    mult := x*y ;
    write ( som , soust , mult , nom , prenom ) ;
    readln ;
end.
```

تبعا لما سبق ذكره أعلاه يمكن استنتاج الطريقة الثانية لكتابة (La Deuxième Syntaxe) التعليمة ***Write*** و التي هي التالية:

Write **(** Liste Des Variables **) ;**

حيث أن:

- Liste Des Variables: تمثل مجموعة المتغيرات المراد أو التي نريد نشر محتواها شريطة أن هذه المجموعة المتغيرات تكون مفصولة بفاصلة. هذه المتغيرات قد تكون من النوع الصحيح أو الحقيقي أو من النوع حرف أو سلسلة أحرف.

المثال 05.07 أدناه نقوم من خلاله بنشر محتوى خمس متغيرات هي: v ,c ,t , z , a , باستخدام تعليمة واحدة فقط بدلا من خمس تعليمات.

مثـــال 05.07:

```
Write ( c , a , z , t , v ) ;
```

إن كل من تعليمات النشر الثلاث التي يتضمنها برنامج المثال 03.07 أعلاه و تعليمة النشر التي يتضمنها برنامج المثال 04.07 أعلاه تسمح بنشر محتوى أو قيم المتغيرات الثلاث *som* ، *soust* ، *mult* ، *nom* ، *prenom* كما يلي

- في سطر واحد.
- و متقاربة كأنها أو على شكل رقم واحد بحيث أنه لا يمكن التمييز بين قيمة كل متغيرة على حدى كما لا يمكن استنتاج قيمة كل متغيرة من هذا الرقم الواحد.

بغرض تجاوز هذا الإشكال الممثل في النشر المتقارب لقيم المتغيرات و الذي يترتب عنه عدم إمكانية معرفة قيمة كل متغيرة على حدى ، هناك تعليمة أخرى تسمح بنشر قيمة كل متغيرة في سطر خاص بها، هذه التعليمة تسمى بالتعليمة ***Writeln*** و التي تشكل موضوع الفقرة الموالية

03.07 – التعليمــة *Writeln* L'instruction

التعليمة ***Writeln*** تسمح بنشر و نسخ و إظهار على الشاشة (Affichage sur écran) محتوى متغيرة ما. طريقة كتابة التعليمة ***Writeln*** La Syntaxe De L'instruction هي التالية:

Writeln **(** *Variable* **) ;**

حيث أن:

- ***Writeln*** : تمثل التعليمة التي تسمح بنشر و نسخ و إظهار على الشاشة محتوى متغيرة.
- **( )** : عبارة عن قوسين.
- *Variable*: تمثل المتغيرة المراد أو التي نريد نشر محتواها. هذه المتغيرة قد تكون من النوع الصحيح أو الحقيقي أو من النوع حرف أو من النوع سلسلة أحرف.
- **;** : تمثل فاصلة منقوطة Point Virgule

من خلال المثال 06.07 أدناه نقوم من خلاله بنشر محتوى المتغيرة *gh* .

مثــال 06.07:

Write (*gh* ***) ;***

و بغرض التوضيح أكثر حول كيفية استخدام التعليمة ***Writeln*** ضمن البرنامج بلغة الـ PASCAL نأخذ المثال التالي:

مثـــال 07.07:

البرنامج أدناه يقوم بنشر على الشاشة محتوى المتغيرة *tot* التي تساوي مجموع محتوى المتغيرتين *a* و *b*.

```
program Affichage_Sur_Ecran ;
var  a , b , tot : integer;
begin
        a := 45000 ;
        b := 50000 ;
        tot := a+b ;
        writeln ( tot  ) ;
        readln ;
end.
```

السطر

```
writeln ( tot  ) ;
```

يسمح بنشر على الشاشة محتوى المتغيرة *tot* كما يلي:

النشر على الشاشة

```
95000
_
```

الكمبيوتر أو الحاسوب يقوم بنشر محتوى المتغيرة *tot* إبتداءا من السطر الأول (Première Ligne) و العمود الأول (Première Colonne) للشاشة و بعدما ينتهي الكمبيوتر من نشر محتوى المتغيرة *tot* الذي هو 95000 ينتقل (_ Le Curseur) إلى السطر الموالي (أي السطر الثاني) حيث يتموضع (_ Le Curseur) في العمود الأول كما هو موضح على الشاشة أعلاه.

برنامج المثال 08.07 أدناه يقوم بنشر محتوى أكثر من متغيرة واحدة

مثـــال 08.07:

```
program Affichage_Sur_Ecran ;
```

```
var  a , b , tot , dif  : integer;
     nom : string ;
begin
        nom := 'ZAHRA' ;
        a := 45000 ;
        b := 5000 ;
        tot := a + b ;
        dif := a – b ;
        writeln ( a  ) ;
        writeln ( b  ) ;
        writeln ( tot  ) ;
        writeln ( dif  ) ;
        writeln ( nom  ) ;
        readln ;
end.
```

البرنامج أعلاه يقوم بنشر ما يلي:

- محتوى المتغيرة *a* الذي يساوي 45000 و هذا إبتداءا من العمود الأول للشاشة إلى غاية العمود الخامس و بعدها ينتقل (_ Le Curseur) إلى السطر الموالي (أي السطر الثاني) حيث يتموضع (_ Le Curseur) في العمود الأول بغرض البدأ في نشر محتوى المتغيرة الموالية الممثلة في *b* .
- محتوى المتغيرة *b* الذي يساوي 5000 و هذا إبتداءا من العمود الأول للشاشة إلى غاية العمود الرابع و بعدها ينتقل (_ Le Curseur) إلى السطر الموالي (أي السطر الثالث) حيث يتموضع (_ Le Curseur) في العمود الأول بغرض البدأ في نشر محتوى المتغيرة الموالية الممثلة في *tot*.
- محتوى المتغيرة *tot* الذي يساوي 50000 و هذا إبتداءا من العمود الأول للشاشة إلى غاية العمود الخامس و بعدها ينتقل (_ Le Curseur) إلى السطر الموالي (أي السطر الرابع) حيث يتموضع (_ Le Curseur) في العمود الأول بغرض البدأ في نشر محتوى المتغيرة الموالية الممثلة في *dif*.
- محتوى المتغيرة *dif* الذي يساوي 40000 و هذا إبتداءا من العمود الأول للشاشة إلى غاية العمود الخامس و بعدها ينتقل (_ Le Curseur) إلى السطر الموالي (أي السطر الخامس)

حيث يتموضع (_ Le Curseur) في العمود الأول بغرض البدأ في نشر محتوى المتغيرة الموالية الممثلة في *nom*.

- محتوى المتغيرة *nom* الذي يساوي ZAHRA و هذا إبتداءا من العمود الأول للشاشة إلى غاية العمود الخامس و بعدها ينتقل (_ Le Curseur) إلى السطر الموالي (أي السطر السادس) حيث يتموضع (_ Le Curseur) في العمود الأول كما هو موضح على الشاشة أدناه:

النشر على الشاشة

45000
5000
50000
40000
ZAHRA
_

04.07 – أشكال النشـر: Les Formats d'Affichage:

يمكن التحكم أو التأثير على كيفية نشر المعطيات بجميع أنواعها

01.04.07 – أشكال نشر المعطيات سلسلة أحرف:

تقوم التعليمتين *Write* و *Writeln* بنشر محتوى المتغيرات من النوع سلسلة أحرف وفقا لطريقة الكتابة التالية:

Writeln **(** *Variable Chaine De Caractères* **) ;**

تبعا لطريقة الكتابة أعلاه، فإن عدد الأماكن (Nombre De Positions) التي يتم فيها نشر محتوى المتغيرة من النوع سلسلة أحرف يساوي طول هذا المحتوى.

مثــال 09.07:

لتكن المتغيرات من النوع سلسلة أحرف *ch_1* ، *ch_2* و *ch_3* حيث أن:

$$ch_1 := 'ZAHRA'$$
$$ch_2 := 'SOUGUEUR'$$
$$ch_3 := 'WILAYA\ \ DE\ \ TIARET'$$

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرات *ch_1* ، *ch_2* و *ch_3*

```
Writeln (ch_1) ;
Writeln (ch_2) ;
Write (ch_3) ;
```

عملية النشر تكون كما يلي:

1. *Writeln (ch_1)* : تقوم بنشر محتوى المتغيرة *ch_1* باستخدام 5 أماكن Cinq Positions (لأن طول المعطية *ZAHRA* التي تمثل محتوى المتغيرة *ch_1* يساوي خمسة) إبتداءا من العمود الأول.
2. *Writeln (ch_2)* : تقوم بنشر محتوى المتغيرة *ch_2* باستخدام 8 أماكن Huit Positions (لأن طول المعطية *SOUGUEUR* التي تمثل محتوى المتغيرة *ch_2* يساوي ثمانية) إبتداءا من العمود الأول.
3. *Write (ch_3)* : تقوم بنشر محتوى المتغيرة *ch_3* باستخدام 16 أماكن Seize Positions (لأن طول المعطية *WILAYA DE TIARET* التي تمثل محتوى المتغيرة *ch_3* يساوي ستة عشر) إبتداءا من العمود الأول.

عند تنفيذ تعليمات النشر الثلاثة أعلاه نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
ZAHRA
SOUGUEUR
WILAYA DE TIARET_
```

ملاحظـة 02.07 :

من خلال طريقة الكتابة أعلاه و المثال 09.07 أعلاه فإن:

- عدد الأماكن (Nombre De Positions) المستخدمة لنشر محتوى متغيرة من النوع سلسلة أحرف **يساوي** طول محتوى هذه المتغيرة.

إضافة إلى طريقة الكتابة أعلاه، هناك طريقة كتابة أخرى كما يلي:

Writeln **(** *Variable Chaine De Caractères* **:** *n* **) ;**

حيث أن:

- n : تمثل عدد الأماكن (Nombre De Positions) التي يتم استخدامها لنشر محتوى المتغيرة من النوع سلسلة أحرف

بغرض توضيح كيفية استخدام طريقة الكتابة الثانية أعلاه، نأخذ المثال التالي:

مثـــال 10.07:

لتكن المتغيرة من النوع سلسلة أحرف ch_1 حيث أن:

$$ch_1:='ZAHRA'$$

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرة سلسلة أحرف ch_1 تبعا لأشكال مختلفة.

```
Writeln (ch_1 : 5)
Writeln (ch_1 : 7) ;
Writeln (ch_1 : 10) ;
Writeln (ch_1 : 15) ;
Writeln (ch_1 : 21) ;
Write (ch_1 : 2) ;
```

عملية النشر تكون كما يلي:

1. *Writeln (ch_1 : 5)* : تقوم بنشر محتوى المتغيرة ch_1 باستخدام 5 أماكن. بما أن طول محتوى المتغيرة يساوي 5، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) الأول دون ترك أي عمود (مكان) فارغ كما هو موضح على الشاشة أدناه:
2. *Writeln (ch_1 : 7)* : تقوم بنشر محتوى المتغيرة ch_1 باستخدام 7 أماكن. بما أن طول محتوى المتغيرة يساوي 5، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) الثالث أي ترك عمودين (مكانين) فارغين كما هو موضح على الشاشة أدناه:
3. *Writeln (ch_1 : 10)* : تقوم بنشر محتوى المتغيرة ch_1 باستخدام 10 أماكن. بما أن طول محتوى المتغيرة يساوي 5، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) السادس أي ترك 5 أعمدة (أماكن) فارغة كما هو موضح على الشاشة أدناه:
4. *Writeln (ch_1 : 15)* : تقوم بنشر محتوى المتغيرة ch_1 باستخدام 15 مكان. بما أن طول محتوى المتغيرة يساوي 5، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) الحادي عشر أي ترك 10 أعمدة (أماكن) فارغة كما هو موضح على الشاشة أدناه:

5. *Writeln (ch_1 : 21)* : تقوم بنشر محتوى المتغيرة *ch_1* باستخدام 21 مكان. بما أن طول محتوى المتغيرة يساوي 5، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) السابع عشر أي ترك 16 عمود (مكان) فارغ كما هو موضح على الشاشة أدناه:

6. *Write (ch_1 : 2)* : تقوم بنشر محتوى المتغيرة *ch_1* باستخدام 2 مكان. بما أن طول محتوى المتغيرة يساوي 5، فإنه لا يمكن نشر المحتوى باستخدام 2 مكان و إنما باستخدام 5 أماكن. عملية النشر سوف تبدأ إبتداءا من العمود (المكان) الأول دون ترك و لا عمود فارغ كما هو موضح على الشاشة أدناه:

النشر على الشاشة

```
ZAHRA
   ZAHRA
      ZAHRA
         ZAHRA
            ZAHRA
ZAHRA_
```

ملاحظـة 03.07 :

طريقة الكتابة التالية:

Writeln (ch_1 : 5)

هي نفسها طريقة الكتابة التالية:

Writeln (ch_1)

لأن طول محتوى المتغيرة *ch_1* يساوي 5

02.04.07 – أشكال نشر المعطيات (القيم) الصحيحة:

تقوم التعليمتين *Write* و *Writeln* بنشر محتوى المتغيرات العددية من النوع الصحيح وفقا لطريقة الكتابة التالية:

Writeln **(** *Variable Numérique De Type Entier* **) ;**

تبعا لطريقة الكتابة أعلاه، فإن عدد الأماكن (Nombre De Positions) التي يتم فيها نشر محتوى المتغيرة العددية من النوع الصحيح يساوي عدد الأرقام المكونة لمحتوى هذه المتغيرة.

مثــال 11.07:

لتكن المتغيرات العددية من النوع الصحيح *val_1* ، *val_2* ، *val_3* و *val_4* حيث أن:

```
val_1 := 5400678
val_2 := 123
val_3 := 98007
val_4 := 4
```

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرات *val_1* ، *val_2* ، *val_3* و *val_4*

```
Writeln (val_1) ;
Writeln (val_2) ;
Writeln (val_3) ;
Write (val_4) ;
```

عملية النشر تكون كما يلي:

1. *Writeln (val_1)* : تقوم بنشر محتوى المتغيرة *val_1* باستخدام 7 أماكن Sept Positions (لأن عدد أرقام المعطية العددية الصحيحة 5400678 التي تمثل محتوى المتغيرة *val_1* يساوي سبعة) إبتداءا من العمود الأول كما هو موضح على الشاشة أدناه:
2. *Writeln (val_2)* : تقوم بنشر محتوى المتغيرة *val_2* باستخدام 3 أماكن Trois Positions (لأن عدد أرقام المعطية العددية الصحيحة 123 التي تمثل محتوى المتغيرة *val_2* يساوي ثلاثة) إبتداءا من العمود الأول كما هو موضح على الشاشة أدناه:
3. *Writeln (val_3)* : تقوم بنشر محتوى المتغيرة *val_3* باستخدام خمسة أماكن Cinq Positions (لأن عدد أرقام المعطية العددية الصحيحة 98007 التي تمثل محتوى المتغيرة *val_3* يساوي خمسة) إبتداءا من العمود الأول كما هو موضح على الشاشة أدناه:
4. *Write (val_4)* : تقوم بنشر محتوى المتغيرة *val_4* باستخدام 3 أماكن Une Position (لأن عدد أرقام المعطية العددية الصحيحة 4 التي تمثل محتوى المتغيرة *val_4* يساوي واحد) إبتداءا من العمود الأول كما هو موضح على الشاشة أدناه:

النشر على الشاشة

```
5400678
123
98007
4_
```

ملاحظـة 04.07 :

من خلال طريقة الكتابة أعلاه و المثال 09.07 أعلاه فإن:

- عدد الأماكن (Nombre De Positions) المستخدمة لنشر محتوى متغيرة عددية من النوع الصحيح **يساوي** عدد الأرقام المكونة لمحتوى هذه المتغيرة.

إضافة إلى طريقة الكتابة أعلاه، هناك طريقة كتابة أخرى كما يلي:

Writeln (*Variable Numérique De Type Entier* **:** *n* **) ;**

حيث أن:

- *n* : تمثل عدد الأماكن (Nombre De Positions) التي يتم استخدامها لنشر محتوى المتغيرة العددية من النوع الصحيح.

بغرض توضيح كيفية استخدام طريقة الكتابة الثانية أعلاه، نأخذ المثال التالي:

مثـــال 12.07:

لتكن المتغيرة العددية من النوع الصحيح val_1 حيث أن:

val_1 := 3450

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرة val_1 تبعا لأشكال مختلفة.

```
Writeln (val_1 : 4) ;
Writeln (val_1 : 5) ;
Writeln (val_1 : 8) ;
Writeln (val_1 : 12) ;
Writeln (val_1 : 20) ;
Writeln (val_1 : 1) ;
```

عملية النشر تكون كما يلي:

1. *Writeln (*val_1 *: 4)* : تقوم بنشر محتوى المتغيرة val_1 باستخدام 4 أماكن. بما أن طول محتوى المتغيرة يساوي 4، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) الأول دون ترك أي عمود (مكان) فارغ كما هو موضح على الشاشة أدناه:
2. *Writeln (*val_1 *: 5)* : تقوم بنشر محتوى المتغيرة val_1 باستخدام 5 أماكن. بما أن طول محتوى المتغيرة يساوي 4، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) الثاني أي ترك عمود (مكان) واحد فارغ كما هو موضح على الشاشة أدناه:

3. *Writeln (*val_1 *: 8)* : تقوم بنشر محتوى المتغيرة val_1 باستخدام 8 أماكن. بما أن طول محتوى المتغيرة يساوي 4، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) الخامس أي ترك 4 أعمدة (أماكن) فارغة كما هو موضح على الشاشة أدناه:
4. *Writeln (*val_1 *: 12)* : تقوم بنشر محتوى المتغيرة val_1 باستخدام 12 مكان. بما أن طول محتوى المتغيرة يساوي 4، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) التاسع أي ترك 8 أعمدة (أماكن) فارغة كما هو موضح على الشاشة أدناه:
5. *Writeln (*val_1 *: 20)* : تقوم بنشر محتوى المتغيرة val_1 باستخدام 20 مكان. بما أن طول محتوى المتغيرة يساوي 4، فإن عملية النشر سوف تبدأ إبتداءا من العمود (المكان) السابع عشر أي ترك 16 عمود (مكان) فارغ كما هو موضح على الشاشة أدناه:
6. *Write (*val_1 *: 1)* : تقوم بنشر محتوى المتغيرة val_1 باستخدام 1 مكان. بما أن طول محتوى المتغيرة يساوي 4، فإنه لا يمكن نشر المحتوى باستخدام 1 مكان و إنما باستخدام 4 أماكن. عملية النشر سوف تبدأ إبتداءا من العمود (المكان) الأول دون ترك و لا عمود فارغ كما هو موضح على الشاشة أدناه:

النشر على الشاشة

```
3450
 3450
    3450
       3450
                3450
3450
```

ملاحظـة 05.07 :

طريقة الكتابة التالية:

Writeln (val_1 : 4)

هي نفسها طريقة الكتابة التالية:

Writeln (val_1)

لأن طول محتوى المتغيرة *val_1* يساوي 4

03.04.07 – أشكال نشر المعطيات (القيم) الحقيقية:

تقوم التعليمتين *Write* و *Writeln* بنشر محتوى المتغيرات العددية من النوع الحقيقي وفقا لطريقة الكتابة التالية:

Writeln **(** *Variable Numérique De Type Réel* **) ;**

بغرض توضيح كيف يتم نشر المعطيات الحقيقية نأخذ المثال التالي:

مثـــال 13.07:

لتكن المتغيرات العددية من النوع الحقيقي r_1 ، r_2 ، r_3 و r_4 حيث أن:

```
r_1 := 4234.1250
r_2 := 123.459
r_3 := 56.89
r_4 := 4.9
```

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرات val_1 ، val_2 ، val_3 و val_4

```
Writeln (r_1) ;
Writeln (r_2) ;
Writeln (r_3) ;
Write (r_4) ;
```

عملية النشر تكون كما هو موضح على الشاشة أدناه:

النشر على الشاشة

```
4.234125000000000E+003
1.234590000000000E+002
5.689000000000000E +001
4.900000000000000E +000
```

تبعا لكل من قيم المتغيرات الحقيقية و كيفية نشر هذه القيم كما هو موضح على الشاشة أعلاه يتضح أن:

- القيم الحقيقية تنشر دائما إبتداءا من العمود الثاني (أي يتم ترك العمود الأول فارغا)
- القيم الحقيقية تنشر برقم واحد فقط قبل الفاصلة كما هو موضح على الشاشة أعلاه
- عدد الأماكن (Nombre De Positions) التي يتم استخدامها لنشر محتوى المتغيرات

العددية من النوع الحقيقي يساوي 23 بما فيها المكان (العمود) الأول الذي يتم تركه فارغا كما هو موضح على الشاشة أعلاه.

إضافة إلى طريقة الكتابة أعلاه، هناك طريقة كتابة أخرى كما يلي:

Writeln **(***Variable Numérique De Type Réel* **:** *n* **) ;**

حيث أن:

- *n* : تمثل عدد الأماكن (Nombre De Positions) التي يتم استخدامها لنشر محتوى المتغيرة العددية من النوع الحقيقي.

بغرض توضيح كيفية استخدام طريقة الكتابة الثانية أعلاه، نأخذ المثال التالي:

مثـــال 14.07:

لتكن المتغيرة العددية من النوع الحقيقي rel_1 حيث أن:

rel_1 := 3455.6789

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرة rel_1 تبعا لأشكال مختلفة.

```
Writeln (rel_1 ) ;
Writeln (rel_1: 23 ) ;
Writeln (rel_1: 22 ) ;
Writeln (rel_1: 21 ) ;
Writeln (rel_1: 20 ) ;
Writeln (rel_1: 19 ) ;
Writeln (rel_1: 15 ) ;
Writeln (rel_1: 30 ) ;
Writeln (rel_1: 10 ) ;
```

عملية النشر تكون كما يلي:

1. (Writeln (rel_1 و (Writeln (rel_1: 23 : كلا التعليمتين تقوم بنشر محتوى المتغيرة العددية من النوع الحقيقي 1_ rel باستخدام 23 مكان بما فيها المكان (العمود) الأول الذي يتم تركه فارغا كما هو موضح على الشاشة أدناه:

2. (Writeln (rel_1: 22 : تقوم بنشر محتوى المتغيرة الحقيقية 1_ rel باستخدام 22 مكان بما فيها المكان (العمود) الأول الذي يتم تركه فارغا. في هذه الحالة سوف يتم الاستغناء عن صفر واحد بالقرب من E كما هو موضح على الشاشة أدناه:

3. (Writeln (rel_1: 21 : تقوم بنشر محتوى المتغيرة الحقيقية rel _1 باستخدام 21 مكان بما فيها المكان (العمود) الأول الذي يتم تركه فارغا. في هذه الحالة سوف يتم الاستغناء عن صفرين اثنين بالقرب من E كما هو موضح على الشاشة أدناه:

4. (Writeln (rel_1: 20 : تقوم بنشر محتوى المتغيرة الحقيقية rel _1 باستخدام 20 مكان بما فيها المكان (العمود) الأول الذي يتم تركه فارغا. في هذه الحالة سوف يتم الاستغناء عن ثلاثة أصفار بالقرب من E كما هو موضح على الشاشة أدناه:

5. (Writeln (rel_1: 19 : تقوم بنشر محتوى المتغيرة الحقيقية rel _1 باستخدام 19 مكان بما فيها المكان (العمود) الأول الذي يتم تركه فارغا. في هذه الحالة سوف يتم الاستغناء عن أربعة أصفار بالقرب من E كما هو موضح على الشاشة أدناه:

6. (Writeln (rel_1: 15 : تقوم بنشر محتوى المتغيرة الحقيقية rel _1 باستخدام 15 مكان بما فيها المكان (العمود) الأول الذي يتم تركه فارغا. في هذه الحالة سوف يتم الاستغناء عن ثمانية (جميع) أصفار بالقرب من E كما هو موضح على الشاشة أدناه:

7. (Writeln (rel_1: 30 : تقوم بنشر محتوى المتغيرة العددية من النوع الحقيقي rel _1 باستخدام 30 مكان. في هذه الحالة سوف يتم نشر محتوى rel _1 ابتداءا من المكان أو العمود التاسع أي ترك ثمانية أعمدة فارغة كما هو موضح على الشاشة أدناه:

8. (Writeln (rel_1: 10 : تقوم بنشر محتوى المتغيرة العددية من النوع الحقيقي rel _1 باستخدام 10 أمكنة كما هو موضح على الشاشة أدناه:

النشر على الشاشة

```
3.455678900000000E+003
3.45567890000000E+003
3.4556789000000E+003
3.455678900000E+003
3.45567890000E+003
3.4556789E+003
        3.455678900000000E+003
3.46E+003
```

إضافة إلى طريقتي الكتابة أعلاه، هناك طريقة كتابة ثالثة أخرى كما يلي:

Writeln **(***Variable Numérique De Type Réel* **:** *n* **:** *m* **) ;**

حيث أن:

- n : تمثل عدد الأماكن التي يتم استخدامها لنشر محتوى المتغيرة العددية من النوع الحقيقي.
- m : تمثل عدد الأرقام بعد الفاصلة.

بغرض توضيح كيفية استخدام طريقة الكتابة الثالثة أعلاه، نأخذ المثال التالي:

مثــال 15.07:

لتكن المتغيرة العددية من النوع الحقيقي rel_1 حيث أن:

rel_1 := 3455.6789

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرة rel_1 تبعا لأشكال مختلفة.

```
Writeln (rel_1: 23 ) ;
Writeln (rel_1: 9 : 4 ) ;
Writeln (rel_1: 10 : 4 ) ;
Writeln (rel_1: 12 : 4 ) ;
Writeln (rel_1: 10 : 5 ) ;
```

عملية النشر تكون كما هو موضح على الشاشة أدناه:

النشر على الشاشة

```
 3.455678900000000E+003
3455.6789
 3455.6789
    3455.6789
3455.67890
```

05.07 – الفرق بين التعليمتين *Write* و *Writeln*

كلا التعليمتين تسمحان بنشر محتوى متغيرة (أو العديد من المتغيرات)، الفرق بينهما يكمن فيما يلي:

- التعليمة ***Write*** تقوم بنشر محتوى متغيرة ما مع **بقـاء** (_ Le Curseur) في نفس السطر الذي تم فيه نشر محتوى المتغيرة و بالضبط في العمود الذي يلي مباشرة عمود انتهاء محتوى المتغيرة التي تم نشر محتواها كما هو موضح على شاشة المثال 16.07 أدناه و بالتالي فإن عملية النشر الموالية سوف تكون في نفس السطر.

مثــال 16.07:

```
program Affichage_Sur_Ecran ;
```

```
var  a  : integer;
begin
        a := 125980 ;
        write ( a  ) ;
        readln ;
end.
```

النشر على الشاشة

```
125980_
```

- التعليمة ***Writeln*** تقوم بنشر محتوى متغيرة ما مع **انتقـال** (_ Le Curseur) إلى السطر الموالي بعد الانتهاء من عملية النشر. كما هو موضح على شاشة المثال 09.07 أدناه و بالتالي فإن عملية النشر الموالية سوف تكون في أو على السطر الموالي.

مثـــال 17.07:

```
program Affichage_Sur_Ecran ;
var  a  : integer;
begin
        a := 125980 ;
        writeln ( a  ) ;
        readln ;
end.
```

النشر على الشاشة

```
125980
_
```

06.07 – التعليمتان *Write* و *Writeln* مـع التعليـق:

Les instructions *Write* et *writeln* avec commentaire:

إضافة إلى أن التعليمتان ***Write*** **و** ***Writeln*** تقومان بنشر محتوى أي نوع من المتغيرات، فإنهما كذالك تقومان بنشر أي تعليق (Commentaire) موضوع بين مزدوجتين (entre deux quottes) أي بين ' ' وفقا لطريقة الكتابة التالية:

```
Write ( ‘ Commentaire ‘ ) ;
Writeln ( ‘ Commentaire ‘ ) ;
```

حيث أن:

- Commentaire: عبارة التعليق الموضوع بين ' ' و الذي هو عبارة عن سلسلة حروف أو أعداد أو رموز أو مزيج من إثنين أو الثلاثة معا.

بغرض التوضيح أكثر نأخذ المثال التالي:

مثـــال 18.07:

لنفرض أننا نود نشر التعاليق التالية:

- iprtjfn,g8ç6_3,é+ ;=[f{&)(rty2
- SETTI HAMID
- Bonjour MONSIEUR
- La Valeur De La Variable X égale :
- Udokf1245*)àé »=La Ville De TIARET
- ComMeNTaIre_°=+:/_____eeeeeee

البرنامج الذي يسمح بنشر التعاليق أعلاه هو التالي:

- البرنامج الأول باستخدام التعليمة ***Writeln***

```
program Affichage_Des_commentaires ;
begin
    writeln ( ‘iprtjfn,g8ç6_3,é+;=[f{&)(rty2’ ) ;
    writeln ( ‘SETTI HAMID’ ) ;
    writeln ( ‘Bonjour MONSIEUR’ ) ;
    writeln ( ‘La Valeur De La Variable X égale :’ ) ;
```

```
writeln ( 'Udokf1245*)àé »=La Ville De TIARET' ) ;
writeln ( 'ComMeNTaIre_°=+:/_____eeeeeee' ) ;
readln ;
end.
```

تنفيذ البرنامج أعلاه يتم كما يلي:

1. يقوم الحاسوب بنشر التعليق الأول **iprtjfn,g8ç6_3,é+;=[f{&)(rty2** في السطر الأول ابتداءا من العمود الأول إلى غاية العمود التاسع و العشرون (لأن التعليق يحتوي على 29) لينتقل _Le Cursseur بعد ذلك إلى السطر الموالي أي الثاني ليبدأ في نشر التعليق الثاني كما هو موضح على الشاشة أدناه:
2. يقوم الحاسوب بنشر التعليق الثاني **SETTI HAMID** في السطر الثاني ابتداءا من العمود الأول إلى غاية العمود الحادي عشر (لأن التعليق يحتوي على 11) لينتقل _Le Cursseur بعد ذلك إلى السطر الموالي أي الثالث ليبدأ في نشر التعليق الثالث كما هو موضح على الشاشة أدناه:
3. يقوم الحاسوب بنشر التعليق الثالث **Bonjour MONSIEUR** في السطر الثالث ابتداءا من العمود الأول إلى غاية العمود السادس عشر (لأن التعليق يحتوي على 16) لينتقل Le _Cursseur بعد ذلك إلى السطر الموالي أي الرابع ليبدأ في نشر التعليق الرابع كما هو موضح على الشاشة أدناه:
4. . . . . .
5. . . . . .
6. يقوم الحاسوب بنشر التعليق السادس **ComMeNTaIre_°=+:/_____eeeeeee** في السطر السادس ابتداءا من العمود الأول إلى غاية العمود التاسع و العشرون (لأن التعليق يحتوي على 29) لينتقل _Le Cursseur بعد ذلك إلى السطر الموالي أي السابع كما هو موضح على الشاشة أدناه:

النشر على الشاشة

```
iprtjfn,g8ç6_3,é+;=[f{&)(rty2
SETTI  HAMID
Bonjour  MONSIEUR
La Valeur De La Variable X égale :
Udokf1245*)àé »=La Ville De TIARET
ComMeNTaIre_°=+:/_____eeeeeee
_
```

- البرنامج الثاني باستخدام التعليمة ***Write***

```
program Affichage_Des_commentaires ;
begin
    write ( 'iprtjfn,g8ç6_3,é+;=[f{&)(rty2' ) ;
    write ( 'SETTI HAMID' ) ;
    write ( 'Bonjour  MONSIEUR' ) ;
    write ( 'La Valeur De La Variable X égale :' ) ;
    write ( 'Udokf1245*)àé »=La Ville De TIARET' ) ;
    write ( 'ComMeNTaIre_°=+:/_____eeeeeee' ) ;
    readln ;
end.
```

تنفيذ البرنامج أعلاه يتم كما يلي:

1. يقوم الحاسوب بنشر التعليق الأول **iprtjfn,g8ç6_3,é+;=[f{&)(rty2** في السطر الأول ابتداءا من العمود الأول إلى غاية العمود رقم 29 (لأن التعليق يحتوي على 29) ثم يتموضع _Le Cursseur بعد ذلك في العمود رقم 30 لنفس السطر أي السطر الأول ليبدأ في نشر التعليق الثاني كما هو موضح على الشاشة أدناه:
2. يقوم الحاسوب بنشر التعليق الثاني **SETTI HAMID** في السطر الأول ابتداءا من العمود رقم 30 إلى غاية العمود رقم 40 (لأن التعليق يحتوي على 11) ليتموضع _Le Cursseur بعد ذلك في العمود رقم 41 دائما في نفس السطر أي السطر الأول ليبدأ في نشر التعليق الثالث كما هو موضح على الشاشة أدناه:
3. يقوم الحاسوب بنشر التعليق الثالث **Bonjour MONSIEUR** في السطر الأول ابتداءا

من العمود رقم 41 إلى غاية العمود رقم 56 (لأن التعليق يحتوي على 16) ثم يتموضع Le _Cursseur بعد ذلك في العمود رقم 57 في نفس السطر أي الأول ليبدأ في نشر التعليق الرابع كما هو موضح على الشاشة أدناه:

4. يقوم الحاسوب بنشر التعليق الرابع **La Valeur De La Variable X égale :** كما يلي:

- نشر الجزء **La Valeur De La Variable** في السطر الأول إبتداءا من العمود 57 إلى غاية العمود 80 و الذي يمثل العمود الأخير للسطر الأول لينتقل Le Cursseur_ بعد ذلك إلى السطر الموالي أي السطر الثاني العمود الأول ليبدأ في نشر الجزء المتبقى من هذا التعليق الرابع كما هو موضح على الشاشة أدناه:
- نشر الجزء **X égale** في السطر الثاني ابتداءا من العمود الأول إلى غاية العمود رقم 8 ثم يتموضع Le Cursseur_ بعد ذلك في العمود رقم 9 في نفس السطر أي الثاني ليبدأ في نشر التعليق الخامس كما هو موضح على الشاشة أدناه:

5. يقوم الحاسوب بنشر التعليق الخامس **Udokf1245*)àé»=La Ville De TIARET :** في السطر الثاني ابتداءا من العمود رقم 9 إلى غاية العمود رقم 41 (لأن التعليق يحتوي على 33) ثم يتموضع Le Cursseur_ بعد ذلك في العمود رقم 42 في نفس السطر أي الثاني ليبدأ في نشر التعليق الخامس كما هو موضح على الشاشة أدناه:

6. يقوم الحاسوب بنشر التعليق السادس **ComMeNTaIre°=+:/_____eeeeeee** في السطر الثاني ابتداءا من العمود رقم 42 إلى غاية العمود رقم 70 (لأن التعليق يحتوي على 29) ليتموضع Le Cursseur_ بعد ذلك في العمود رقم 71 لنفس السطر أي الثاني كما هو موضح على الشاشة أدناه:

النشر على الشاشة

iprtjfn,g8ç6_3,é+;=[f{&)(rty2SETTI HAMIDBonjour MONSIEURLa Valeur De La Variable X égale :Udokf1245*)àé »=La Ville De TIARETComMeNTaIre_°=+:/_____eeeeeee _

مثـــال 19.07:

لنفرض أننا نود الحصول على الشكل التالي على الشاشة

النشر على الشاشة

```
Nom       PRENOM      NOTE_1   NOTE_2   NOTE_3   MOYENNE
_____________________________________________________
SEDDI     ALI         15.50    13.75    18.25    15.83
```

البرنامج الذي يسمح بالحصول على الشكل أعلاه هو التالي:

```
program Affichage ;
begin
writeln ( 'Nom     PRENOM     NOTE_1   NOTE_2   NOTE_3   MOYENNE' ) ;
writeln ( '_____________________________________________________' ) ;
writeln ( 'SEDDI   ALI        15.50    13.75    18.25    15.83' ) ;
readln ;
end.
```

مثـــال 20.07:

لنفرض أننا نود الحصول على الشكل التالي على الشاشة.

النشر على الشاشة

```
NOM : BOUCHA
PRENOM : MOHAMED
DATE DE NAISSANCE : 17 / 03 / 1970
LIEU DE NAISSANCE : TIARET
SITUATION FAMILIALLE : CELIBATAIRE
NIVEAU SCOLAIRE : 4 EME ANNEE MOYENNE
ADRESSE :  SANS ADRESSE
---------------------------------------
```

البرنامج الذي يسمح بذلك هو التالي:

```
program Affichage ;
begin
    writeln ( 'NOM : BOUCHA' ) ;
```

```
    writeln ( 'PRENOM : MOHAMED' ) ;
    writeln ( 'DATE DE NAISSANCE : 17 / 03 / 1970' ) ;
    writeln ( 'LIEU DE NAISSANCE : TIARET' ) ;
    writeln ( 'SITUATION FAMILIALLE : CELIBATAIRE' ) ;
    writeln ( 'NIVEAU SCOLAIRE : 4 EME ANNEE MOYENNE' ) ;
    writeln ( 'ADRESSE :  SANS ADRESSE' ) ;
    writeln ( '----------------------------------------' ) ;
    readln ;
end.
```

07.07 – التعليمـة *Writeln* و قفـز (تخطـي) السطـر :

L'instruction *Writeln* et le saut de ligne :

إضافة إلى أن التعليمة ***Writeln*** تقوم بنشر محتوى أي نوع من المتغيرات و بنشر أي تعليق، فإنها كذالك تقوم **بتخطي أو قفز سطر واحد** (Saut Une Ligne) عند إدراجها وحدها في البرنامج وفقا لطريقة الكتابة التالية:

```
Writeln ;
```

بغرض التوضيح أكثر نأخذ المثال التالي:

مثـــال 21.07:

ليكن البرنامج أدناه و الذي من خلاله نقوم بنشر محتوى 5 متغيرات، كل محتوى في سطر دون تخطي أو قفز سطر واحد بعد كل عملية نشر.

```
program Affichage ;
var  a , b , c , som , diff , mult , divi , care : integer ;
begin
    a := 5 ;
    b := 625 ;
    c := 100 ;
    som := a + b ;
    diff := b – a ;
```

```
        mult := a*c ;
        divi := c div a ;
        care := sqr(a) ;
        writeln ( ' La Somme De a et b Egale : ' , som ) ;
        writeln (' La Difference Entre b et a Egale : ' , diff ) ;
        writeln (' Le Produit De a et c Egale : ' , mult ) ;
        writeln (' Le Resultat De c Sur a Egale : ' , divi ) ;
        writeln (' Le carree De a Egale : ', care ) ;
        readln ;
end.
```

نتيجة عملية النشر تظهر على الشاشة كما يلي:

النشر على الشاشة

```
La Somme De a et b Egale : 630
La Difference Entre b et a Egale : 620
Le Produit De a et c Egale : 500
Le Resultat De c Sur a Egale : 20
Le carree De a Egale : 390625
_
```

بغرض تخطي سطر واحد بعد كل عملية نشر أي ترك سطر واحد فارغ نقوم بإضافة أو إدراج التعليمة ***writeln*** بعد كل تعليمة نشر كما يوضحه البرنامج أدناه:

```
program Affichage ;
var  a , b , c , som , diff , mult , divi , care : integer ;
begin
        a := 5 ;
        b := 625 ;
        c := 100 ;
        som := a+b ;
        diff := b-a ;
        mult := a*c ;
        divi := c div a ;
```

```
care := sqr(a) ;
writeln ( ' La Somme De a et b Egale : ' , som ) ;
writeln ;
writeln (' La Difference Entre b et a Egale : ' , diff ) ;
writeln ;
writeln (' Le Produit De a et c Egale : ' , mult ) ;
writeln ;
writeln (' Le Resultat De c Sur a Egale : ' , divi ) ;
writeln ;
writeln (' Le carree De a Egale : ', care ) ;
writeln ;
readln ;
end.
```

نتيجة عملية النشر تظهر على الشاشة كما يلي:

النشر على الشاشة

```
La Somme De a et b Egale : 630

La Difference Entre b et a Egale : 620

Le Produit De a et c Egale : 500

Le Resultat De c Sur a Egale : 20

Le carree De a Egale : 390625
_
```

ملاحظـة 06.07 :

- بغرض تخطي سطرين أي ترك سطرين فارغين نقوم بإدراج التعليمة ***writeln*** مرتين كما يلي:

writeln ;

writeln ;

- بغرض تخطي ثلاثة أسطر أي ترك ثلاثة أسطر فارغة نقوم بإدراج التعليمة 3 مرات كما يلي:

writeln ;

writeln ;

writeln ;

- و هكذا . . .

08.07 – التعليمتان *Write* و *Writeln* و نشـر نتائـج العمليـات الحسابيـة:

Les instructions *Write* et *writeln* et l'affichage des résultats des opérations arithmétiques:

إضافة إلى أن التعليمتان ***Write*** **و** ***Writeln*** تقومان بنشر محتوى أي نوع من المتغيرات و بنشر أي تعليق، فإنهما كذالك تقومان بنشر **نتيجـة أي عمليـة حسابيـة** (Opération Arithmétique) وفقا لطريقة الكتابة التالية:

Write **(** Opération Arithmétique **) ;**

Writeln **(** Opération Arithmétique **) ;**

حيث أن:

- Opération Arithmétique: عبارة عن عملية حسابية من العمليات الحسابية التي تم التطرق لها في الفصل الرابع المتعلق بمعالجة المعطيات.

بغرض التوضيح أكثر نأخذ المثال التالي:

مثـــال 22.07:

لنفرض أنه لدينا المتغيرات الصحيحة التالية:

$a = 5$

$b = 625$

$c = 100$

و نود نشر نتائج العمليات التالية:

- $a+b$
- $b-c$
- $a*c$
- c/a
- a^2
- $\sqrt{b}$
- $\sqrt{\frac{(b-a)^2}{100}}$

البرنامج الذي يسمح بذلك هو التالي:

```
program Affichage_Des_Résultats_Des_Opérations ;
var  a , b , c : integer ;
begin
        a := 5 ;
        b := 625 ;
        c := 100 ;
        write ( a + b ) ;
        write ( b – c ) ;
        write ( a * c ) ;
        write ( c / a ) ;
        write ( sqr(a) ) ;
        write ( sqrt(b) ) ;
        write ( sqrt( sqr(b – a)  div  100 ) ) ;
        readln ;
end.
```

ملاحظـة 07.07 :

التعليمات الستة المتعلقة بنشر نتائج العمليات الحسابية الست يمكن تعويضها و استبدالها بتعليمة واحدة فقط كما يلي:

```
write ( a + b , b – c , a * c , c / a , sqr(a) , sqrt(b) , sqrt( sqr(b – a)  div  100 ) ) ;
```

و عليه يصبح برنامج المثال 15.07 أعلاه كما يلي:

```
program Affichage_Des_Résultats_Des_Opérations ;
var  a , b , c : integer ;
begin
    a := 5 ;
    b := 625 ;
    c := 100 ;
    write ( a+b , b-c , a*c , c/a , sqr(a) , sqrt(b) , sqrt( sqr(b-a) div 100 ) ) ;
    readln ;
end.
```

بما أن التعليمة ***write*** تقوم بنشر نتائج العمليات الحسابية في سطر واحد و بالقرب من بعضها البعض مما يترتب عنه عدم إمكانية التمييز و التفرقة بين نتائج كل عملية على حدى نقوم باستخدام التعليمة ***Writeln*** التي تقوم بنشر كل نتيجة في سطر و عليه يصبح البرنامج الأول للمثال 15.07 أعلاه كما يلي:

```
program Affichage_Des_Résultats_Des_Opérations ;
var  a , b , c : integer ;
begin
    a := 5 ;
    b := 625 ;
    c := 100 ;
    writeln ( a+b ) ;
    writeln ( b-c ) ;
    writeln ( a*c ) ;
    writeln ( c/a ) ;
    writeln ( sqr(a) ) ;
    writeln ( sqrt(b) ) ;
    writeln ( sqrt( sqr(b-a) div 100 ) ) ;
    readln ;
```

end.

مما سبق ذكره أعلاه، يمكن استنتاج الطريقة الثانية لكتابة (La Deuxième Syntaxe) التعليمتان ***write*** و ***Writeln*** في حالة نشر نتائج العمليات الحسابية.

Write **(** Opérations Arithmétiques **) ;**

Writeln **(** Opérations Arithmétiques **) ;**

حيث أن:

Opérations Arithmétiques: عبارة عن مجموعة من العملية الحسابية مفصولة فيما بينها بفواصل (Des Opérations Arithmétiques Séparées Par Des Virgules).

09.07 – التعليمتان *Write* و *Writeln* و نشـر كل من التعاليـق و محتوى المتغيرات و نتائـج العمليـات الحسابيـة:

Les instructions *Write* et *writeln* et l'affichage des commentaires et le contenu des variables et les résultats des opérations arithmétiques:

من الفقرات السابقة تم التطرق إلى أن ***Write*** و ***Writeln*** تقومان بما يلي:

- نشر **محتوى أي نوع من المتغيرات.**
- نشر **التعاليق.**
- نشر **نتائـج العمليات الحسابيـة.**

إضافة إلى ذلك يمكن نشر اثنين معا كما يلي:

- تعليق و محتوى متغيرة.
- تعليق و نتيجة عملية حسابية.

بغرض التوضيح أكثر نأخذ المثال التالي:

مثـــال 23.07:

لنفرض أنه لدينا المتغيرات الصحيحة التالية:

$a = 15000$

$b = 2000$

$c = 5000$

و نود نشر قيم هذه المتغيرات على الشاشة كما يلي:

```
La Valeur De La Variable a Egale : 15000
_
```

انطلاقا مما هو منشور على الشاشة نلاحظ أنه متكون من جزئين هما التعليق و محتوى المتغيرة *a* :

1. التعليق الذي هو **La Valeur De La Variable a Egale :** و الذي سوف يتم نشره باستخدام التعليمة ***Write*** وفقا لطريقة الكتابة التالية:

Write **(** ‘ La Valeur De La Variable a Egale : ‘ **) ;**

وفقا لهذه التعليمة سوف يقوم الحاسوب بنشر التعليق **La Valeur De La Variable a Egale :** و بعده يتموضع _Le Cursseur في العمود الموالي لعمود نهاية التعليق و في نفس سطر نشر التعليق ليبدأ في نشر محتوى المتغيرة *a*.

2. محتوى المتغيرة *a* الذي هو **15000** و الذي سوف يتم نشره باستخدام التعليمة ***Write*** وفقا لطريقة الكتابة التالية:

Write **(** a **) ;**

أو من الأحسن باستخدام التعليمة ***Writeln*** وفقا لطريقة الكتابة التالية:

Writeln **(** a **) ;**

باستخدام التعليمة ***Writeln*** يقوم الحاسوب بنشر محتوى المتغيرة *a* في نفس سطر التعليق و إبتداءا من العمود الموالي لعمود نهاية التعليق ثم ينتقل _Le Cursseur بعد ذلك إلى السطر الموالي على عكس ***Write*** **(** a **)** التي لا ينتقل _Le Cursseur إلى السطر الموالي بعد نشر محتوى المتغيرة *a* بل يبقى في العمود الموالي لعمود نهاية القيمة 15000 التي تمثل محتوى المتغيرة *a*.

ملاحظـة 08.07 :

يمكن تعويض و استبدال التعليمتين

Write **(** ‘ La Valeur De La Variable a Egale : ‘ **) ;**

Writeln **(** a **) ;**

بتعليمة واحدة فقط تسمح في آن واحد بنشر التعليق و محتوى المتغيرة *a* كما يلي:

Write **(** ‘ La Valeur De La Variable a Egale : ‘ **,** *a* **) ;**

أو

Writeln **(** ‘ La Valeur De La Variable a Egale : ‘ **,** *a* **) ;**

بذلك نستنتج طريقة الكتابة (La Syntaxe) للتعليمة ***Writeln*** و التعليمة ***Write*** عندما تستخدمان في نشر تعليق و محتوى متغيرة في آن واحد.

Write **(** ' Commentaire ' **,** Variable **) ;**

Writeln **(** ' Commentaire ' **,** Variable **) ;**

حيث أن:

- Commentaire: عبارة عن تعليق موضوع بين ' '.
- Variable: إسم المتغيرة المراد نشر محتواها.
- , : فاصلة تفصل التعليق عن إسم المتغيرة.

مثـــال 24.07:

لتكن لدينا نفس معطيات المثال 23.07 أعلاه:

و نود نشر على الشاشة ما يلي:

```
La Valeur De La Variable a Egale : 15000
La Valeur De La Variable b Egale : 2000
La Valeur De La Variable c Egale : 5000
La Somme De a et c Egale         : 20000
Le Resultat De (a+b+c)/2  Egale  : 11000
Le Carré De b Egale              : 4000000
La Racine Carré De a Egale       : 122.474
c  à La Puissance 3 Egale        : 125  10^9
_
```

البرنامج الذي يسمح بالحصول على ذلك هو التالي:

```
program Affichage_Commentaire_et_Contenu_Variable ;
var  a , b , c : integer ;
     s , m , ca , ra , cub : real ;
begin
     a := 15000 ;
     b := 2000 ;
     c := 5000 ;
     s := a+c ;
     m := (a+b+c)/2 ;
```

```
ca := sqr(b) ;
ra := sqrt(a) ;
cub := c**3 ;
writeln ( ' La Valeur De La Variable a Egale : ' , a ) ;
writeln ( ' La Valeur De La Variable b Egale : ' , b ) ;
writeln ( ' La Valeur De La Variable c Egale : ' , c ) ;
writeln ( ' La Somme De a et c Egale : ' , s ) ;
writeln ( ' Le Resultat De (a+b+c)/2 Egale : ' , m ) ;
writeln ( ' Le Carré De b Egale : ' , ca ) ;
writeln ( ' La Racine Carré De a Egale : ' , ra ) ;
writeln ( ' c à La Puissance 3 Egale : ' , cub ) ;
readln ;
end.
```

إضافة إلى أن التعليمتين و يمكن لهما نشر تعليق و محتوى متغيرة، فإنه كذالك يمكن لهما نشر تعليق و نتيجة عملية حسابية.

بغرض التوضيح أكثر نأخذ المثال التالي:

مثـــال 25.07:

لنفرض أننا نود الحصول على نفس ما تم الحصول عليه في شاشة المثال 17.07 أعلاه دون الاستعانة بالمتغيرات من النوع الحقيقي *s , m , ca , ra , cub*

البرنامج الذي يسمح بذلك هو التالي:

```
program Affichage_Commentaire_et_Resultat_Opération ;
var a , b , c : integer ;
begin
    a := 15000 ;
    b := 2000 ;
    c := 5000 ;
    writeln ( ' La Valeur De La Variable a Egale : ' , a ) ;
    writeln ( ' La Valeur De La Variable b Egale : ' , b ) ;
    writeln ( ' La Valeur De La Variable c Egale : ' , c ) ;
```

```
writeln ( ' La Somme De a et c Egale      : ' , a+c ) ;
writeln ( ' Le Resultat De (a+b+c)/2  Egale  : ' , (a+b+c)/2  ) ;
writeln ( ' Le Carré De b Egale           : ' , sqr(b)  ) ;
writeln ( ' La Racine Carré De a Egale    : ' , sqrt(a) ) ;
writeln ( ' c  à La Puissance 3 Egale     : ' , c**3 ) ;
readln  ;
end.
```

بذلك نستنتج طريقة الكتابة (La Syntaxe) للتعليمة ***Writeln*** و التعليمة ***Write*** عندما تستخدمان في نشر تعليق و نتيجة عملية حسابية في آن واحد.

Write **(** ‘ Commentaire ‘ **,** Opération Arithmétique **) ;**

Writeln **(** ‘ Commentaire ‘ **,** Opération Arithmétique **) ;**

حيث أن:

- Commentaire: عبارة عن تعليق موضوع بين ' '.
- Opération Arithmétique: عملية حسابية.
- , : فاصلة تفصل التعليق عن العملية الحسابية.

تمـاريـن الفصـل

✍ التمــريــن 01.07:

1. قدم مختلف التعليمات التي تسمح بنشر و نسخ على الشاشة محتوى المتغيرات.

2. قدم مختلف طرق كتابة (Les Différentes Syntaxes) كل تعليمة نشر متوصل إليها في السؤال الأول.

✍ التمــريــن 02.07:

1. قدم الفرق بين التعليمتين *Write* و *Writeln*.

✍ التمــريــن 03.07:

1. قدم طريقة الكتابة (La Syntaxe) للتعليمة ***Write*** في حالة استخدامها لنشر محتوى متغيرة مهما كان نوعها.

✍ التمــريــن 04.07:

لتكن x متغيرة من النوع الصحيح و التي تساوي ما يلي:

$$x = 12350$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرة x باستخدام التعليمة ***Write***.

✍ التمــريــن 05.07:

لتكن y متغيرة من النوع الحقيقي و التي تساوي ما يلي:

$$y = 4500.345$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرة y باستخدام التعليمة ***Write***.

✍ التمــريــن 06.07:

لتكن z متغيرة من النوع سلسلة أحرف و التي تساوي ما يلي:

z= ‘La Ville De Tiaret Est Très Belle’

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرة z باستخدام التعليمة ***Write***.

✍ التمــريــن 07.07:

لتكن a متغيرة من النوع حرف و التي تساوي ما يلي:

$$a = 'Z'$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرة a باستخدام التعليمة ***Write***.

✍ التمــريــن 08.07:

1. أكتب البرنامج الذي يسمح بنشر محتوى متغيرة من اقتراحك هذا باستخدام التعليمة ***Write***.

✍ التمــريــن 09.07:

1. قدم طريقة الكتابة (La Syntaxe) للتعليمة ***Writeln*** في حالة استخدامها لنشر محتوى متغيرة مهما كان نوعها.

✍ التمــريــن 10.07:

لتكن x متغيرة من النوع الصحيح و التي تساوي ما يلي:

$$x = 9000$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرة x باستخدام التعليمة ***Writeln*** .

✍ التمــريــن 11.07:

لتكن y متغيرة من النوع الحقيقي و التي تساوي ما يلي:

$$y = 0.000045$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرة y باستخدام التعليمة ***Writeln*** .

✍ التمــريــن 12.07:

لتكن z متغيرة من النوع سلسلة أحرف و التي تساوي ما يلي:

z= ‘Mon Rêve Est De Visiter La Wilaya De Tiaret’

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرة z باستخدام التعليمة ***Writeln*** .

✍ التمــريــن 13.07:

لتكن a متغيرة من النوع حرف و التي تساوي ما يلي:

$$a = 'Z'$$

- **المطلــوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرة a باستخدام التعليمة ***Writeln*** .

✍ التمــريــن 14.07:

1. أكتب البرنامج الذي يسمح بنشر محتوى متغيرة من اقتراحك هذا باستخدام التعليمة ***Writeln***

✍ التمــريــن 15.07:

1. حدد الفرق بين التعليمتين *Write* و *Writeln*.

✍ التمــريــن 16.07:

لتكن المعطيات التالية:

$$a = 35740$$
$$b = 259004$$
$$c = 34000$$
$$d = 50$$

- **المطلــوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرات (المعطيات) أعلاه في سطر واحد.

✍ التمــريــن 17.07:

1. أكتب البرنامج الذي يسمح بنشر محتوى متغيرات (المعطيات) التمرين 03.07 أعلاه بحيث كل محتوى متغيرة في سطر خاص به.

2. توضيح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 18.07:

1. أكتب البرنامج الذي يسمح بنشر محتوى متغيرات (المعطيات) التمرين 03.07 أعلاه كما يلي:

- محتوى المتغيرتين a و c في سطر واحد.
- محتوى المتغيرتين b و d في سطر واحد آخر.

2. توضيح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمـريـن 19.07:

1. أكتب البرنامج الذي يسمح بنشر محتوى متغيرات (المعطيات) التمرين 03.07 أعلاه كما يلي:

- محتوى المتغيرتين a و c و d في سطر واحد.
- محتوى المتغيرتين b في سطر آخر.

2. توضيح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمـريـن 20.07:

لتكن المعطيات التالية:

$$a = 1250$$
$$b = 2478$$
$$p = 'SIHEM'$$
$$d_n = '17/03/1988'$$
$$l_n = 'TIARET'$$
$$niv = '2\ Eme\ Année\ Commerce'$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر محتوى المتغيرات (المعطيات) أعلاه في سطر واحد.

2. توضيح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمـريـن 21.07:

1. أكتب البرنامج الذي يسمح بنشر محتوى متغيرات (المعطيات) التمرين 03.07 أعلاه بحيث كل محتوى متغيرة في سطر خاص به.

2. توضيح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمـريــن 22.07:

1. أكتب البرنامج الذي يسمح بنشر محتوى متغيرات (المعطيات) التمرين 03.07 أعلاه كما يلي:

- محتوى المتغيرتين a و n_d في سطر واحد.
- محتوى المتغيرتين b و n_l في سطر واحد آخر.
- محتوى المتغيرتين p و niv في سطر واحد آخر.

2. توضيح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمـريــن 23.07:

2. قدم طريقة الكتابة (La Syntaxe) للتعليمة ***Write*** التي تسمح بنشر تعليق معين.

✍ التمـريــن 24.07:

ليكن التعليق التالي:

Nom : CHABANI OMAR

- **المطلـوب:**

2. أكتب البرنامج الذي يسمح بنشر التعليق أعلاه و هذا باستخدام التعليمة ***Write***.

✍ التمـريــن 25.07:

2. أكتب البرنامج الذي يسمح بنشر تعليق من اقتراحك و هذا باستخدام التعليمة ***Write***.

✍ التمـريــن 26.07:

1. قدم طريقة الكتابة (La Syntaxe) للتعليمة ***Writeln*** التي تسمح بنشر تعليق معين.

✍ التمـريــن 27.07:

ليكن التعليق التالي:

Prénom : AIT FERHAT MOURAD

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر التعليق أعلاه و هذا باستخدام التعليمة ***Writeln*** .

✍ التمــريــن 28.07:

3. أكتب البرنامج الذي يسمح بنشر تعليق من اقتراحك و هذا باستخدام التعليمة ***Writeln***

✍ التمــريــن 29.07:

لنفرض أننا نود نشر التعاليق التالية:

- iprtjfn,g8ç6_3,é+ ;=[f{&)(rty2
- SETTI HAMID
- Bonjour MONSIEUR
- La Valeur De La Variable X égale :
- Udokf1245*)àé »=La Ville De TIARET
- ComMeNTaIre_°=+:/_____eeeeeee

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر التعاليق أعلاه في سطر واحد.

2. توضيح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 30.07:

لنفرض أنه لدينا تعاليق التمرين 29.07 أعلاه:

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر التعاليق أعلاه بحيث كل تعليق يتم نشره سطر خاص به.

2. توضيح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج المتوصل إليه في السؤال الأول.

✍ التمــريــن 31.07:

لنفرض أننا نود الحصول على الشكل التالي على الشاشة

Nom	PRENOM	NOTE_1	NOTE_2	NOTE_3	MOYENNE
SEDDI	ALI	15.50	13.75	18.25	15.83

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالحصول على الشكل أعلاه

✍ التمـريـن 32.07:

لنفرض أننا نود الحصول على الشكل التالي على الشاشة.

```
NOM : BOUCHA
PRENOM : MOHAMED
DATE DE NAISSANCE : 17 / 03 / 1970
LIEU DE NAISSANCE : TIARET
SITUATION FAMILIALLE : CELIBATAIRE
NIVEAU SCOLAIRE : 4 EME ANNEE MOYENNE
ADRESSE :  SANS ADRESSE
-------------------------------------
```

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالحصول على الشكل أعلاه

✍ التمـريـن 33.07:

1. أكتب البرنامج الذي يسمح بالحصول على الشكل المبين على الشاشة أدناه:

```
          I-/ Introduction Des Données :
Donnez La Valeur De La Variable Entière a :
Donnez La Valeur De La Variable Réelle  b :
Donnez La Valeur De La Variable String  n :
          II-/ Affichage Des Données :
La Valeur De a Egale :
La Valeur De b Egale :
La Valeur De n Egale :
_
```

✍ التمـريــن 34.07:

1. أكتب البرنامج الذي يسمح بالحصول على الشكل المبين على الشاشة أدناه

```
NOM :
PRENOM :
Date De Naissance :
Lieu De Naissance :
Note Mathématique :
Note Informatique :
Note Statistique :
Note Comptabilité :
_
```

✍ التمـريــن 35.07:

1. قدم طريقة الكتابة (La Syntaxe) للتعليمة ***Write*** في حالة استخدامها لنشر في آن واحد تعليق و محتوى متغيرة.

2. قدم طريقة الكتابة (La Syntaxe) للتعليمة ***Writeln*** في حالة استخدامها لنشر في آن واحد تعليق و محتوى متغيرة.

✍ التمـريــن 36.07:

لتكن المعطيات التالية:

$a = 3500$

$b = 45000$

$x = 25000$

نود الحصول على الشكل أدناه على الشاشة.

```
La Valeur De La Variable b Egale : 45000
```

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالحصول على الشكل أعلاه

✍ التمــريــن 37.07:

لتكن المعطيات التالية:

$d = 2500$

$e = 5600$

$f = 8000$

نود الحصول على الشكل أدناه على الشاشة.

```
La Valeur De La Variable d Egale : 2500
La Valeur De La Variable e Egale : 5600
La Valeur De La Variable f Egale : 8000
_
```

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالحصول على الشكل أعلاه

✍ التمــريــن 38.07:

لتكن المعطيات التالية:

nom='SETTI HAMID'

حيث نود الحصول على الشكل أدناه على الشاشة.

```
Votre Nom Et Prenom : SETTI HAMID
MERCI
```

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالحصول على الشكل أعلاه

✍ التمــريــن 39.07:

لنفرض أنه لدينا المتغيرات التالية:

$x = 15$

$y = 20$

$z = 50$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر قيم المتغيرات $x, y, z, a, a1, a2, a3, b, b1, b2, m, c$ مع تعليق يعبر عن ماذا تمثل قيمة متغيرة حيث أن:

- $a = x + y$
- $a1 = x + z$
- $a2 = y + z$
- $a3 = x - y$
- $b = x - z$
- $b1 = y - z$
- $b2 = x + y + z$
- $m = \frac{(x + y + z)}{3}$
- $c = sqr(20)$

✍ التمـريـن 40.07:

لنفرض أنه لدينا المتغيرات الصحيحة التالية:

$a = 15000$

$b = 2000$

$c = 5000$

و نود نشر على الشاشة ما يلي:

```
La Valeur De La Variable a Egale : 15000
La Valeur De La Variable b Egale : 2000
La Valeur De La Variable c Egale : 5000
La Somme De a et c Egale         : 20000
Le Resultat De (a+b+c)/2  Egale  : 11000
Le Carré De b Egale              : 4000000
La Racine Carré De a Egale       : 122.474
c  à La Puissance 3 Egale        : 125  10^9
_
```

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالحصول على الشكل أعلاه

✍ التمـريـن 41.07:

لتكن المتغيرة من النوع سلسلة أحرف ch_1 حيث أن:

ch_1:='ZAHRA'

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرة سلسلة أحرف *ch_1* تبعا لأشكال مختلفة.

```
Writeln (ch_1 : 5)
Writeln (ch_1 : 7) ;
Writeln (ch_1 : 10) ;
Writeln (ch_1 : 15) ;
Writeln (ch_1 : 21) ;
Write (ch_1 : 2) ;
```

- **المطلــوب:**

1. وضح ماذا يتم الحصول على الشاشة عند تنفيذ تعليمات النشر أعلاه.

✍ التمــريــن 42.07:

لتكن المتغيرة العددية من النوع الصحيح val_1 حيث أن:

val_1 := 3450

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرة val_1 تبعا لأشكال مختلفة.

```
Writeln (val_1 : 4) ;
Writeln (val_1 : 5) ;
Writeln (val_1 : 8) ;
Writeln (val_1 : 12) ;
Writeln (val_1 : 20) ;
Writeln (val_1 : 1) ;
```

- **المطلــوب:**

1. وضح ماذا يتم الحصول على الشاشة عند تنفيذ تعليمات النشر أعلاه.

✍ التمــريــن 43.07:

لتكن المتغيرات العددية من النوع الحقيقي *r_1* ، *r_2* ، *r_3* و *r_4* حيث أن:

```
r_1 := 4234.1250
r_2 := 123.459
r_3 := 56.89
r_4 := 4.9
```

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرات val_1 ، val_2 ، val_3 و val_4

```
Writeln (r_1) ;
Writeln (r_2) ;
Writeln (r_3) ;
Write (r_4) ;
```

- **المطلـوب:**

1. وضح ماذا يتم الحصول على الشاشة عند تنفيذ تعليمات النشر أعلاه.

✍ التمـريـن 44.07:

لتكن المتغيرة العددية من النوع الحقيقي rel_1 حيث أن:

rel_1 := 3455.6789

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرة rel_1 تبعا لأشكال مختلفة.

```
Writeln (rel_1 ) ;
Writeln (rel_1: 23 ) ;
Writeln (rel_1: 22 ) ;
Writeln (rel_1: 21 ) ;
Writeln (rel_1: 20 ) ;
Writeln (rel_1: 19 ) ;
Writeln (rel_1: 15 ) ;
Writeln (rel_1: 30 ) ;
Writeln (rel_1: 10 ) ;
```

- **المطلـوب:**

1. وضح ماذا يتم الحصول على الشاشة عند تنفيذ تعليمات النشر أعلاه.

✍ التمـريـن 44.07:

لتكن المتغيرة العددية من النوع الحقيقي rel_1 حيث أن:

rel_1 := 3455.6789

التعليمات التالية تقوم بنشر على الشاشة محتوى المتغيرة rel_1 تبعا لأشكال مختلفة.

```
Writeln (rel_1: 23 ) ;
Writeln (rel_1: 9 : 4 ) ;
Writeln (rel_1: 10 : 4 ) ;
```

```
Writeln (rel_1: 12 : 4 ) ;
Writeln (rel_1: 10 : 5 ) ;
```

- **المطلـوب:**

1. وضح ماذا يتم الحصول على الشاشة عند تنفيذ تعليمات النشر أعلاه.

✍ التمـريـن 45.07:

ليكن البرنامج أدناه و الذي من خلاله نقوم بنشر محتوى 5 متغيرات، كل محتوى في سطر دون تخطي أو قفز سطر واحد بعد كل عملية نشر.

```
program Affichage ;
var  a , b , c , som , diff , mult , divi , care : integer ;
begin
        a := 5 ;
        b := 625 ;
        c := 100 ;
        som := a + b ;
        diff := b – a ;
        mult := a * c ;
        divi := c div a ;
        care := sqr(a) ;
        writeln ( ' La Somme De a et b Egale : ' , som ) ;
        writeln (' La Difference Entre b et a Egale : ' , diff ) ;
        writeln (' Le Produit De a et c Egale : ' , mult ) ;
        writeln (' Le Resultat De c Sur a Egale : ' , divi ) ;
        writeln (' Le carree De a Egale : ', care ) ;
        readln ;
end.
```

- **المطلـوب:**

1. وضح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج أعلاه.

التمـريــن 46.07:

ليكن البرنامج أدناه و الذي من خلاله نقوم بنشر محتوى 5 متغيرات، كل محتوى في سطر مع تخطي أو قفز سطر واحد بعد كل عملية نشر.

```
program Affichage ;
var  a , b , c , som , diff , mult , divi , care : integer ;
begin
        a := 5 ;
        b := 625 ;
        c := 100 ;
        som := a+b ;
        diff := b–a ;
        mult := a*c ;
        divi := c div a ;
        care := sqr(a) ;
        writeln ( ' La Somme De a et b Egale : ' , som ) ;
        writeln ;
        writeln (' La Difference Entre b et a Egale : ' , diff ) ;
        writeln ;
        writeln (' Le Produit De a et c Egale : ' , mult ) ;
        writeln ;
        writeln (' Le Resultat De c Sur a Egale : ' , divi ) ;
        writeln ;
        writeln (' Le carree De a Egale : ', care ) ;
        writeln ;
        readln ;
end.
```

- **المطلـوب:**

1. وضح ماذا يتم الحصول على الشاشة عند تنفيذ البرنامج أعلاه.

✍ التمـريــن 47.07:

لنفرض أنه لدينا المتغيرات الصحيحة التالية:

$$a = 5$$
$$b = 625$$
$$c = 100$$

و نود نشر نتائج العمليات التالية:

- $a + b$
- $b - c$
- $a * c$
- c / a
- a^2
- $\sqrt{b}$
- $\sqrt{\frac{(b-a)^2}{100}}$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بنشر العمليات الحسابية أعلاه.

✍ التمـريــن 48.07:

لنفرض أنه لدينا المتغيرات الصحيحة التالية:

$$a = 15000$$
$$c = 5000$$
$$b = 2000$$

و نود نشر قيم هذه المتغيرات على الشاشة كما يلي:

```
La Valeur De La Variable a Egale : 15000
La Valeur De La Variable b Egale : 2000
La Valeur De La Variable c Egale : 5000
_
```

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالحصول على النشر الموضح على الشاشة أعلاه.

التمـريــن 49.07:

لتكن معطيات التمرين 48.07 أعلاه و نود نشر على الشاشة ما يلي:

```
La Valeur De La Variable a Egale : 15000
La Valeur De La Variable b Egale : 2000
La Valeur De La Variable c Egale : 5000
La Somme De a et c Egale         : 20000
Le Resultat De (a+b+c)/2  Egale  : 11000
Le Carré De b Egale              : 4000000
La Racine Carré De a Egale       : 122.474
c  à La Puissance 3 Egale        : 125  10^9
_
```

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بالحصول على النشر الموضح على الشاشة أعلاه.

التمـريــن 50.07:

الجدول أدناه يقدم خمسة قيم للمتغير Y

قيم المتغير Y
$Y_1 = 5$
$Y_2 = 2$
$Y_3 = 1$
$Y_4 = 2$
$Y_5 = 3$

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال المعطيات أعلاه كما هو موضح على الشاشة أدناه

2. حساب نتائج العمليات التالية:

- مربع القيم أي: $Y_1^2, Y_2^2, Y_3^2, Y_4^2, Y_5^2$
- مكعب القيم أي: $Y_1^3, Y_2^3, Y_3^3, Y_4^3, Y_5^3$
- القيم مرفوعة قوة أربعة أي: $Y_1^4, Y_2^4, Y_3^4, Y_4^4, Y_5^4$
- مجموع قيم المتغير Y و الذي يساوي: $SOM1 = Y_1 + Y_2 + Y_3 + Y_4 + Y_5$
- مجموع مربع قيم المتغير Y و الذي يساوي:

$$SOM2 = Y_1^2 + Y_2^2 + Y_3^2 + Y_4^2 + Y_5^2$$

- مجموع مكعب قيم المتغير Y و الذي يساوي:

$$SOM3 = Y_1^3 + Y_2^3 + Y_3^3 + Y_4^3 + Y_5^3$$

- مجموع القيم مرفوعة قوة أربعة أي: $SOM4 = Y_1^4 + Y_2^4 + Y_3^4 + Y_4^4 + Y_5^4$

3. حساب المتوسط الحسابي La Moyenne Arithmétique $\overline{Y}$ للمتغير Y و الذي يساوي:

$$Moy_Ar = \frac{Y_1 + Y_2 + Y_3 + Y_4 + Y_5}{5}$$

4. حساب المتوسط الهندسي La Moyenne Géométrique للمتغير Y و الذي يساوي:

$$Moy_Ge = \sqrt[5]{Y_1 \times_2 \times Y_3 \times Y_4 \times Y_5}$$

5. حساب التباين La Variance $Var(Y)$ للمتغير Y و الذي يساوي:

$$Var(Y) = \frac{\left(Y_1 - \overline{Y}\right)^2 + \left(Y_2 - \overline{Y}\right)^2 + \left(Y_3 - \overline{Y}\right)^2 + \left(Y_4 - \overline{Y}\right)^2 + \left(Y_5 - \overline{Y}\right)^2}{5}$$

6. حساب العزم البسيط من الدرجة الأولى M_S^1 للمتغير Y حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 1\ : M_S^1 = \frac{Y_1 + Y_2 + Y_3 + Y_4 + Y_5}{5}$$

7. حساب العزم البسيط من الدرجة الثانية M_S^2 للمتغير Y حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 2\ : M_S^2 = \frac{Y_1^2 + Y_2^2 + Y_3^2 + Y_4^2 + Y_5^2}{5}$$

8. حساب العزم البسيط من الدرجة الثالثة M_S^3 للمتغير Y حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 3\ : M_S^3 = \frac{Y_1^3 + Y_2^3 + Y_3^3 + Y_4^3 + Y_5^3}{5}$$

9. حساب العزم البسيط من الدرجة الرابعة M_S^4 للمتغير Y حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 4\ : M_S^4 = \frac{Y_1^4 + Y_2^4 + Y_3^4 + Y_4^4 + Y_5^4}{5}$$

10. حساب العزم المركزي من الدرجة الأولى M_C^1 للمتغير Y حيث أن:

$$M_C^1 = \frac{\left(Y_1 - \overline{Y}\right)^1 + \left(Y_2 - \overline{Y}\right)^1 + \left(Y_3 - \overline{Y}\right)^1 + \left(Y_4 - \overline{Y}\right)^1 + \left(Y_5 - \overline{Y}\right)^1}{5}$$

11. حساب العزم المركزي من الدرجة الثانية M_C^2 للمتغير Y حيث أن:

$$M_C^2 = \frac{\left(Y_1 - \overline{Y}\right)^2 + \left(Y_2 - \overline{Y}\right)^2 + \left(Y_3 - \overline{Y}\right)^2 + \left(Y_4 - \overline{Y}\right)^2 + \left(Y_5 - \overline{Y}\right)^2}{5}$$

12. حساب العزم المركزي من الدرجة الثالثة M_C^3 للمتغير Y حيث أن:

$$M_C^3 = \frac{\left(Y_1 - \overline{Y}\right)^3 + \left(Y_2 - \overline{Y}\right)^3 + \left(Y_3 - \overline{Y}\right)^3 + \left(Y_4 - \overline{Y}\right)^3 + \left(Y_5 - \overline{Y}\right)^3}{5}$$

13. نشر نتائج العمليات الحسابية كما هو موضح على الشاشة أدناه

I-/ Introduction Des Valeurs De La Variable Y :

Donnez La Valeur De $Y1$:
Donnez La Valeur De $Y2$:
Donnez La Valeur De $Y3$:
Donnez La Valeur De $Y4$:
Donnez La Valeur De $Y5$:

II-/ Affichage Des Résultats :

1) La Somme Des Valeurs De La Variable Y Egale :
2) La Somme Des Valeurs De Y à La Puissance 2 Egale :
3) La Somme Des Valeurs De Y à La Puissance 3 Egale :
4) La Somme Des Valeurs De Y à La Puissance 4 Egale :
5) La Moyenne Arithmétique De La Variable Y Egale :
6) La Moyenne Géométrique De La Variable Y Egale :
7) La Variance De La Variable Y Egale :
8) Moment Simple D ordre 1 Egale :
9) Moment Simple D ordre 2 Egale :
10) Moment Simple D ordre 3 Egale :
11) Moment Simple D ordre 4 Egale :
12) Moment Centré D ordre 1 Egale :
13) Moment Centré D ordre 2 Egale :
14) Moment Centré D ordre 3 Egale :

–

✍ التمـريـن 51.07:

الجدول أدناه يقدم أربع قيم للمتغير X و كذا التكرارت الموافقة لها

قيم المتغير X	التكرار الموافق لكل قيمة
$X_1 = 2$	$n_1 = 3$
$X_2 = 4$	$n_2 = 2$
$X_3 = 3$	$n_3 = 5$
$X_4 = 6$	$n_4 = 2$

– **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال المعطيات أعلاه كما هو موضح على الشاشة أدناه

2. حساب نتائج العمليات التالية:

- مجموع التكرارات الذي يساوي: $N = n_1 + n_2 + n_3 + n_4$
- مجموع قيم المتغير X و الذي يساوي: $SOM1 = X_1 + X_2 + X_3 + X_4$
- جداء القيم بتكراراتها أي حساب: $X_1 \times n_1, X_2 \times n_2, X_3 \times n_3, X_4 \times n_4$
- مجموع جداء القيم بتكراراتها أي حساب:

$$SOM11 = X_1 \times n_1 + X_2 \times n_2 + X_3 \times n_3 + X_4 \times n_4$$

- مربع قيم المتغير X أي حساب: $X_1^2, X_2^2, X_3^2, X_4^2$
- مجموع مربع قيم المتغير X و الذي يساوي: $SOM2 = X_1^2 + X_2^2 + X_3^2 + X_4^2$
- مجموع جداء مربع القيم بتكراراتها أي حساب:

$$SOM22 = X_1^2 \times n_1 + X_2^2 \times n_2 + X_3^2 \times n_3 + X_4^2 \times n_4$$

3. حساب المتوسط الحسابي La Moyenne Arithmétique للمتغير X و الذي يساوي:

$$M_AR = \frac{X_1 \times n_1 + X_2 \times n_2 + X_3 \times n_3 + X_4 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

4. حساب المتوسط الهندسي La Moyenne Géométrique للمتغير X و الذي يساوي:

$$Moy_Ge = \sqrt[(n1+n2+n3+n4)]{X_1^{n1} \times X_2^{n2} \times X_3^{n3} \times X_4^{n4}}$$

5. حساب التباين La Variance للمتغير X و الذي يساوي ما يلي:

$$VAR(X) = \frac{\left(X_1 - \overline{X}\right)^2 \times n_1 + \left(X_2 - \overline{X}\right)^2 \times n_2 + \left(X_3 - \overline{X}\right)^2 \times n_3 + \left(X_4 - \overline{X}\right)^2 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

6. حساب العزم البسيط من الدرجة الأولى M_S^1 للمتغير X حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 1 : M_S^1 = \frac{X_1^1 \times n_1 + X_2^1 \times n_2 + X_3^1 \times n_3 + X_4^1 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

7. حساب العزم البسيط من الدرجة الثانية M_S^2 للمتغير X حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 2 : M_S^2 = \frac{X_1^2 \times n_1 + X_2^2 \times n_2 + X_3^2 \times n_3 + X_4^2 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

8. حساب العزم البسيط من الدرجة الثالثة M_S^3 للمتغير X حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 3 : M_S^3 = \frac{X_1^3 \times n_1 + X_2^3 \times n_2 + X_3^3 \times n_3 + X_4^4 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

9. حساب العزم البسيط من الدرجة الرابعة M_S^4 للمتغير X حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 4 : M_S^4 = \frac{X_1^4 \times n_1 + X_2^4 \times n_2 + X_3^4 \times n_3 + X_4^4 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

10. حساب العزم المركزي من الدرجة الأولى M_C^1 للمتغير X حيث أن:

$$M_C^1 = \frac{\left(X_1 - \overline{X}\right)^1 \times n_1 + \left(X_2 - \overline{X}\right)^1 \times n_2 + \left(X_3 - \overline{X}\right)^1 \times n_3 + \left(X_4 - \overline{X}\right)^1 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

11. حساب العزم المركزي من الدرجة الثانية M_C^2 للمتغير X حيث أن:

$$M_C^2 = \frac{\left(X_1 - \overline{X}\right)^2 \times n_1 + \left(X_2 - \overline{X}\right)^2 \times n_2 + \left(X_3 - \overline{X}\right)^2 \times n_3 + \left(X_4 - \overline{X}\right)^2 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

12. حساب العزم المركزي من الدرجة الثالثة M_C^3 للمتغير X حيث أن:

$$M_C^3 = \frac{\left(X_1 - \overline{X}\right)^3 \times n_1 + \left(X_2 - \overline{X}\right)^3 \times n_2 + \left(X_3 - \overline{X}\right)^3 \times n_3 + \left(X_4 - \overline{X}\right)^3 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

13. حساب العزم المركزي من الدرجة الرابعة M_C^4 للمتغير X حيث أن:

$$M_C^4 = \frac{\left(X_1 - \overline{X}\right)^4 \times n_1 + \left(X_2 - \overline{X}\right)^4 \times n_2 + \left(X_3 - \overline{X}\right)^4 \times n_3 + \left(X_4 - \overline{X}\right)^4 \times n_4}{n_1 + n_2 + n_3 + n_4}$$

14. نشر المعطيات التي تم إدخالها كما هو موضح على الشاشة أدناه

15. نشر نتائج العمليات التي تم حسابها كما هو موضح على الشاشة أدناه

I-/ Introduction Des Valeurs De La Variable X :

Donnez La Valeur De $X1$:
Donnez La Valeur De $n1$:
Donnez La Valeur De $X2$:
Donnez La Valeur De $n2$:
Donnez La Valeur De $X3$:
Donnez La Valeur De $n3$:
Donnez La Valeur De $X4$:
Donnez La Valeur De $n4$:

II-/ Affichage Des Valeurs De La Variable X :

La Valeur De $X1$ Egale :
La Valeur De $X2$ Egale :
La Valeur De $X3$ Egale :
La Valeur De $X4$ Egale :

III-/ Affichage Des Résultats :

Xi	ni	X_i^2	$X_i \times ni$	$X_i^2 \times ni$
2	3	4	6	12
4	2	16	.	.
3	5	9	.	.
6	2	36	.	.
15	**12**	**65**	**.**	**.**

a) La Moyenne Arithmétique De La Variable X Egale :
b) La Moyenne Géométrique De La Variable X Egale :
c) La Variance De La Variable X Egale :
d) Moment Simple D ordre 1 Egale :
e) Moment Simple D ordre 2 Egale :
f) Moment Simple D ordre 3 Egale :
g) Moment Simple D ordre 4 Egale :
h) Moment Centré D ordre 1 Egale :
i) Moment Centré D ordre 2 Egale :
j) Moment Centré D ordre 3 Egale :
k) Moment Centré D ordre 4 Egale :

الفصل الثامن

التعليمات الشرطية

Les Instructions Conditionnelles

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- تعليمة القفز اللاشرطي *GOTO*.
- طريقة كتابة (La Syntaxe) تعليمة القفز اللاشرطي *GOTO*.
- تعليمة التصريح بالعلامات (Les Etiquettes) *LABEL* .
- طريقة كتابة (La Syntaxe) تعليمة التصريح بالعلامات (Les Etiquettes) *LABEL* .
- التعليمة الشرطية IF….THEN….ELSE….
- طريقة كتابة التعليمة الشرطية IF….THEN….ELSE….
- التعليمة المركبة
- التعليمات IF المتداخلة
- تعليمة التوجيه المتعدد CASE….OF….
- طريقة كتابة تعليمة التوجيه المتعدد CASE….OF….

01.08 – تمهيد:

يقوم الحاسوب بتنفيذ (Exécution) أي برنامج بلغة الـ PASCAL وفقا لمبدأ التسلسل، أي أن تعليمات أي برنامج تنفذ تسلسليا أي تصاعديا بدءا بالتعليمة الأولى (السطر الأول) ثم التعليمة الثانية (السطر الثاني) إلى غاية آخر تعليمة (آخر سطر). يمكن التدخل في عملية التنفيذ التسلسلي لتعليمات البرنامج بحيث لا يقوم الحاسوب بتنفيذ أسطر البرنامج تسلسليا و إنما يمكن له تخطي أو قفز سطر (تعليمة) أو مجموعة من الأسطر (التعليمات). عملية التدخل هذه تتم بالاستعانة بتعليمات خاصة سوف يتم التطرق إليها من خلال هذا الفصل.

02.08 – تعليمة القفز اللاشرطي GOTO :

L'instruction De Branchement Inconditionnel :

يقوم الحاسوب بتنفيذ (Exécution) أي برنامج بلغة الـ PASCAL وفقا لمبدأ التسلسل، أي أن تعليمات أي برنامج تنفذ تسلسليا أي تصاعديا بدءا بالتعليمة الأولى ثم التعليمة الثانية إلى غاية آخر تعليمة. بغرض توضيح مبدأ التسلسل في تنفيذ التعليمات التي يتكون منها أي برنامج، نأخذ المثال التالي:

مثـال 01.08:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program XXXX ;
Var  a, b, c ,d : integer ;
begin
    a:= 1500 ;
    b:= 5000 ;
    c:= a+b ;
    d:= b-a ;
    Writeln (a) ;
    Writeln (b) ;
    Writeln (c) ;
    Writeln (d) ;
end.
```

يتم تنفيذ البرنامج أعلاه تسلسليا (تعليمة بعد تعليمة) كما يلي:

1. السطر رقم 1 : *program XXXX;*

2. السطر رقم 2 : *Var a, b, c ,d : integer ;* التصريح بأربع متغيرات من النوع الصحيح.
3. السطر رقم 3 : *begin*
4. السطر رقم 4 : *a:= 1500 ;* توجيه القيمة الصحيحة 1500 إلى المتغيرة *a*.
5. السطر رقم 5 : *b:= 5000 ;* توجيه القيمة الصحيحة 5000 إلى المتغيرة *b*.
6. السطر رقم 6 : *c:= a+b ;* القيام بعملية جمع محتوى المتغيرتين *a* و *b* و توجيه النتيجة إلى المتغيرة *c*.
7. السطر رقم 7 : *d:= b-a ;* القيام بعملية طرح محتوى المتغيرة *a* من محتوى المتغيرة *b* و توجيه النتيجة إلى المتغيرة *d*.
8. السطر رقم 8 : *Writeln (a) ;* نشر محتوى المتغيرة *a*.
9. السطر رقم 9 : *Writeln (b) ;* نشر محتوى المتغيرة *b*.
10. السطر رقم 10 : *Writeln (c) ;* نشر محتوى المتغيرة *c*.
11. السطر رقم 11 : *Writeln (d) ;* نشر محتوى المتغيرة *d*.
12. السطر رقم 12 : *end.*

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
1500
5000
6500
3500
_
```

ملاحظة 01.08:

تبعا لمبدأ التسلسل في تنفيذ تعليمات برنامج ما من قبل الحاسوب، فإنه لا يمكن أن يترك أو يتخطى أي تعليمة دون أن ينفذها أي بدون تنفيذ، بمعنى أن الحاسوب يقوم بتنفيذ جميع تعليمات البرنامج تسلسليا.

يمكن التأثير على مبدأ التسلسل في تنفيذ التعليمات، بحيث لا يتم تنفيذ جميع التعليمات تسلسليا. عملية التأثير على مبدأ التسلسل تكون أو تأخذ أحد الأشكال التالية:

- عدم تنفيذ تعليمة من تعليمات البرنامج أي تخطي (قفز) تعليمة دون تنفيذها.

- عدم تنفيذ أكثر من تعليمة واحدة من تعليمات البرنامج أي تخطي (قفز) مجموعة من التعليمات دون تنفيذها.
-

تتم عملية التأثير على مبدأ التسلسل في تنفيذ التعليمات عن طريق تخطي تعليمة أو مجموعة من التعليمات باستخدام **تعليمة القفز اللاشرطي GOTO**، وفقا لطريقة الكتابة التالية:

GOTO *étiquette* ***;***

حيث أن:

- ***GOTO*** : تمثل تعليمة القفز اللاشرطي و التي تعني **الذهاب إلى**
- *étiquette* : و التي تعني علامة، هذه الأخيرة عبارة عن عدد صحيح. هذا العدد الصحيح يكتب قبل و في نفس سطر التعليمة المراد الذهاب إليها بواسطة التعليمة *GOTO* أعلاه.
- **;** : فاصلة منقوطة

ملاحظة 02.08:

يشترط التصريح بالعلامة (*étiquette*) في البرنامج في الجزء الأول الخاص بالتصريح. تتم عملية التصريح بالعلامة باستخدام التعليمة ***LABEL*** وفقا لطريقة الكتابة التالية:

LABEL *étiquette* ***;***

أما إذا كانت لدينا أكثر من علامة (*étiquette*) واحدة أي العديد من العلامات فإنه يتم التصريح بها وفقا لطريقة الكتابة التالية:

LABEL *étiquette_1* **,** *étiquette_2* **,** *étiquette_3* **,** *étiquette_4* ***;***

بغرض التوضيح كيفية التأثير على مبدأ التسلسل في تنفيذ التعليمات باستخدام التعليمة GOTO، نأخذ المثال التالي:

مثال 02.08:

ليكن برنامج المثال 01.08 أعلاه و لنفرض أننا لا نريد تنفيذ تعليمة السطر رقم 8 (*Writeln (a) ;*) بمعنى أن الحاسوب يقوم بتنفيذ البرنامج بدءا بتعليمة السطر رقم 1 إلى غاية تعليمة

السطر رقم 7 ثم يقوم الحاسوب بعد ذلك بتخطي (قفز) تعليمة السطر رقم 8 ليواصل الحاسوب تنفيذ تعليمة السطر رقم 9 إلى غاية تعليمة السطر رقم 11.

حتى نتمكن من عدم تنفيذ السطر رقم 8 و ذلك عن طريق تخطيه نقوم بالاستعانة بتعليمة القفز اللاشرطي GOTO

حيث يتم إدراج و إضافة هذه التعليمة مباشرة قبل السطر الذي لا نريد تنفيذه و هو السطر رقم 8 كما يلي:

GOTO 1 ;

حيث أن:

- 1: تمثل *étiquette* و التي يتم كتابتها قبل و في نفس سطر التعليمة المراد تنفيذها أي السطر رقم 9 كما يلي:

1 Writeln (c) ;

و بما أن أي *étiquette* مستخدمة يجب التصريح بها في البرنامج في الجزء الأول الخاص بالتصريح، فإنه يتم التصريح بـ l'*étiquette* 1 كما يلي:

LABEL 1 ;

بإضافة التعليمتين أعلاه نحصل على البرنامج التالي:

```
program XXXX ;
Var  a, b, c ,d : integer ;
LABEL  1 ;
begin
    a:= 1500 ;
    b:= 5000 ;
    c:= a+b ;
    d:= b-a ;
    GOTO 1 ;
    Writeln (a) ;
    1 : Writeln (b) ;
    Writeln (c) ;
    Writeln (d) ;
end.
```

انطلاقا من البرنامج أعلاه، فإن التعليمة الوحيدة التي لا يتم تنفيذها هي:

Writeln (a) ;

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
5000
6500
3500
_
```

مثـال 03.08:

ليكن برنامج المثال 01.08 أعلاه و لنفرض أننا لا نريد تنفيذ تعليمات الأسطر التالية:

- السطر رقم 8 ; *Writeln (a)*
- السطر رقم 9 ; *Writeln (b)*
- السطر رقم 10 ; *Writeln (c)*

حتى نتمكن من عدم تنفيذ الأسطر أعلاه نقوم بإضافة و إدراج تعليمة القفز اللاشرطي GOTO مباشرة قبل السطر رقم 8 كما يلي:

GOTO 5 ;

حيث أن:

- 5: تمثل *étiquette* و التي يتم كتابتها في السطر الذي يلي السطر 8 أي السطر 9 كما يلي:

5 : Writeln (d) ;

و بما أن أي *étiquette* مستخدمة يجب التصريح بها في البرنامج في الجزء الأول الخاص بالتصريح، فإنه يتم التصريح بـ *l'étiquette 5* كما يلي:

LABEL 5 ;

بإضافة التعليمتين أعلاه نحصل على البرنامج التالي:

```
program XXXX ;
Var  a, b, c ,d : integer ;
LABEL 5;
begin
```

```
    a:= 1500 ;
    b:= 5000 ;
    c:= a+b ;
    d:= b-a ;
    GOTO 5 ;
    Writeln (a) ;
    Writeln (b) ;
    Writeln (c) ;
  5 : Writeln (d) ;
end.
```

مثـال 04.08:

ليكن برنامج المثال 01.08 أعلاه و لنفرض أننا لا نريد تنفيذ تعليمات الأسطر التالية:

- السطر رقم 7 ; *d:= b-a*
- السطر رقم 8 ; *Writeln (a)*
- السطر رقم 9 ; *Writeln (b)*
- السطر رقم 10 ; *Writeln (c)*
- السطر رقم 11 ; *Writeln (d)*

يتم عدم تنفيذ تعليمات الأسطر أعلاه عن طريق إضافة التعليمة ***goto*** قبل السطر رقم 7 كما هو موضح في البرنامج أدناه:

```
program XXXX ;
Var  a, b, c ,d : integer ;
Label 3 ;
begin
    a:= 1500 ;
    b:= 5000 ;
    c:= a+b ;
    goto 3 ;
    d:= b-a ;
    Writeln (a) ;
```

```
Writeln (b) ;
Writeln (c) ;
Writeln (d) ;
3: end.
```

مثال 05.08:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program ZZZ ;
var x,y,z : real ;
label 2 ;
begin
 writeln (' LA VILLE  DE  SOUGUEUR  EST  TRES  BELLE ' ) ;
Goto 2 ;
x:= 5.6 ;
y:= 1.2 ;
z:=x/y ;
Writeln (x) ;
Writeln (y) ;
Writeln (z) ;
2 : Writeln (' Merci De Votre Visite') ;
Writeln (' Merci De Votre Visite') ;
end.
```

الأسطر التي لا يتم تنفيذها من قبل الحاسوب هي التالية:

```
x:= 5.6 ;
y:= 1.2 ;
z:=x/y ;
Writeln (x) ;
Writeln (y) ;
Writeln (z) ;
```

عند تنفيذ البرنامج أعلاه نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
LA VILLE DE SOUGUEUR EST TRES BELLE
Merci De Votre Visite
Merci De Votre Visite
_
```

ملاحظة 03.08:

تسمح تعليمة القفز اللاشرطي بالانتقال إلى أي سطر من أسطر البرنامج بما فيها السطر الأخير (*end*) و ما عدا الأسطر التالية:

- الأسطر المتواجدة في الجزء الخاص بالتصريح بالمتغيرات
- السطر الذي يلي مباشرة التعليمة *goto*

03.08 – التعليمة الشرطية IF…THEN…ELSE:

L'instruction Conditionnelle IF…THEN…ELSE :

في بعض المرات نقوم بتنفيذ تعليمة أو مجموعة من التعليمات إذا تحقق شرط (Une Condition) معين و إذا لم يتحقق هذا الشرط نقوم بتنفيذ تعليمة أو مجموعة من التعليمات الأخرى تختلف عن الأولى أي تختلف عن تلك التعليمات في حالة تحقق الشرط. بغرض توضيح ما سبق نورد الأمثلة التالية:

مثال 06.08:

نقوم بإدخال عن طريق لوحة المفاتيح معطيتين عدديتين (صحيحتين أو حقيقيتين) a و b حيث:

إذا كان $a > b$ نقوم بالنشر على الشاشة ما يلي:

a **Est Supérieure A** *b*

أما إذا لم يكن (أي إذا a يختلف عن b) نقوم بنشر على الشاشة ما يلي:

a **Est Inferieure A** *b*

في هذه الحالة:

يتمثل الشرط (La condition) في:

$$a > b$$

هذا الشرط قد يكون محقق (صحيح) أو غير محقق (غير صحيح).

إذا كان هذا الشرط محقق، نقوم بتنفيذ تعليمة ممثلة في نشر على الشاشة: *a* **Est Supérieure A** *b*

يتم ترجمة هذه التعليمة بلغة الـ PASCAL كما يلي:

Writeln (‘*a* **Est Supérieure A** *b*’)

إذا لم يكن هذا الشرط محقق، نقوم بتنفيذ تعليمة ممثلة بنشر على الشاشة: *a* **Est Inferieure A** *b*

يتم ترجمة هذه التعليمة بلغة الـ PASCAL كما يلي:

Writeln (‘*a* **Est Inferieure A** *b*’) ;

مثـال 07.08:

نقوم بإدخال عن طريق لوحة المفاتيح معطيتين عدديتين (صحيحتين أو حقيقيتين) *a* و *b* حيث:

إذا كان *a* = *b* نقوم بالنشر على الشاشة ما يلي:

a **Egale A** *b*

أما إذا لم يكن نقوم بنشر على الشاشة ما يلي:

a **Différent De** *b*

في هذه الحالة:

يتمثل الشرط (La condition) في:

a* = *b

هذا الشرط قد يكون محقق (صحيح) أو غير محقق (غير صحيح).

إذا كان هذا الشرط محقق، نقوم بتنفيذ تعليمة ممثلة في نشر على الشاشة: *a* **Egale A** *b*

يتم ترجمة هذه التعليمة بلغة الـ PASCAL كما يلي:

Writeln (‘*a* **Egale A** *b*’)

إذا لم يكن هذا الشرط محقق، نقوم بتنفيذ تعليمة ممثلة بنشر على الشاشة: *a* **Différent De** *b*

يتم ترجمة هذه التعليمة بلغة الـ PASCAL كما يلي:

Writeln (‘*a* **Différent De** *b*’) ;

مثـال 08.08:

يمكن أن يكون الشرط كما يلي:

- *a* < *b* و الذي يعني *a* أقل من *b*
- *a* <> *b* و الذي يعني *a* لا يساوي *b* أي *a* يختلف عن *b*
- *a* <= *b* و الذي يعني *a* أقل أو يساوي *b*
- *a* >= *b* و الذي يعني *a* أكبر أو يساوي *b*

مثـال 09.08:

نقوم بإدخال عن طريق لوحة المفاتيح معطية عددية و لتكن *x* حيث:

إذا كان $x > 150$ نقوم بالنشر على الشاشة ما يلي:

x **Est Supérieure A** 150

أما إذا لم يكن نقوم بنشر على الشاشة ما يلي:

x **Est Inferieure A** 150

في هذه الحالة يتمثل الشرط في: $x > 150$ و الذي قد يكون محقق أو غير محقق.

إذا كان هذا الشرط محقق، نقوم بتنفيذ التعليمة التالية:

Writeln (‘x **Est Supérieure A** 150’)

إذا لم يكن هذا الشرط محقق، نقوم بتنفيذ التعليمة التالية:

Writeln (‘x **Est Inferieure A** 150’) ;

مثـال 10.08:

نقوم بإدخال عن طريق لوحة المفاتيح 3 علامات و لتكن $note_1, note_2, note_3$ ثم نقوم بحساب المعدل $moy = \frac{note_1 + note_2 + note_3}{3}$ حيث:

إذا كان المعدل أقل من 10 أي $moy >= 10$ نقوم بالنشر على الشاشة ما يلي:

L’étudiant Admis

أما إذا لم يكن نقوم بنشر على الشاشة ما يلي:

L’étudiant Ajourné

في هذه الحالة يتمثل الشرط في: $moy >= 10$ و الذي قد يكون محقق أو غير محقق.

إذا كان هذا الشرط محقق، نقوم بتنفيذ التعليمة التالية:

Writeln (‘ L’étudiant Admis ’)

إذا لم يكن هذا الشرط محقق، نقوم بتنفيذ التعليمة التالية:

Writeln (‘ L’étudiant Ajourné’) ;

مثـال 11.08:

نقوم بإدخال عن طريق لوحة المفاتيح الأجر الذي يتلقاه موظف ما و الذي نسميه *sal* ثم نقوم بحساب مقدار الضريبة *imp* التي يدفعها هذا الموظف و التي تحسب كما يلي:

إذا كان الأجر يساوي أو يقل عن 50000 دج أي $sal <= 50000$ فإن مقدار الضريبة *imp* يساوي:

$$imp = 0.25 \times sal$$

أما إذا لم يكن فإن مقدار الضريبة *imp* يساوي:

$$imp = 0.30 \times sal$$

في هذه الحالة يتمثل الشرط في: $sal <= 50000$ و الذي قد يكون محقق أو غير محقق.

إذا كان هذا الشرط محقق، نقوم بتنفيذ التعليمة التالية:

*Imp := 0.25 * sal ;*

إذا لم يكن هذا الشرط محقق، نقوم بتنفيذ التعليمة التالية:

*Imp := 0.30 * sal ;*

من خلال الأمثلة أعلاه تم التعرف على ما يلي:

- كيف يكون الشرط
- التعليمة التي تنفذ في حالة **تحقق هذا الشرط**
- التعليمة التي تنفذ في حالة **عدم تحقق هذا الشرط**

تكتب التعليمة الشرطية ....IF....THEN....ELSE وفقا لطريقة الكتابة التالية:

```
If Condition Then Instruction_1
          Else  Instruction_2 ;
```

حيث أن:

- **If** : و التي تعني بالعربية **إذا** أما بالفرنسية فتعني **Si**
- Condition : و التي تعني شرط ، هذا الأخير يكون في أغلب الحالات عبارة عن مقارنة بين قيمتين أو أكثر مثلما هو موضح في الأمثلة أعلاه
- **Then** : و التي تعني بالعربية **إذن** أما بالفرنسية فتعني **Alors**
- **Else** : و التي تعني بالعربية **و إلا** أما بالفرنسية فتعني **Si Non**
- Instruction_1 : عبارة عن تعليمة يتم تنفيذها في حالة **تحقق** الشرط Condition. هذه التعليمة يمكن لها أن تكون:
  - تعليمة نشر تعليق أو نشر محتوى متغيرة.
  - تعليمة معالجة معطيات.
  - تعليمة إدخال معطيات.
  - تعليمة القفز اللاشرطي *goto*.

  مثلما توضحه الأمثلة أدناه
- Instruction_2 : عبارة عن تعليمة يتم تنفيذها في حالة **عدم تحقق** الشرط Condition. هذه التعليمة يمكن لها أن تكون:
  - تعليمة نشر تعليق أو نشر محتوى متغيرة.

- تعليمة معالجة معطيات.
- تعليمة إدخال معطيات.
- تعليمة القفز اللاشرطي *goto*.

مثلما توضحه الأمثلة أدناه

يتم تنفيذ التعليمة ....IF....THEN....ELSE أعلاه كما يلي:

يقوم الحاسوب باختبار صحة الشرط Condition حيث:

- إذا كان صحيحا يقوم بتنفيذ التعليمة Instruction_1
- إذا كان غير صحيحا يقوم بتنفيذ التعليمة Instruction_2

بغرض توضيح كيفية استخدام التعليمة الشرطية ....IF....THEN....ELSE أعلاه نأخذ الأمثلة التالية:

مثـال 12.08:

يتم استخدام التعليمة الشرطية ....IF....THEN....ELSE للمثال 06.08 أعلاه كما يلي:

If *a > b* **Then** *writeln ('a Est Supérieure A b ')*

Else *writeln ('a Est Inferieure A b ') ;*

يتم تنفيذ التعليمة ....IF....THEN....ELSE أعلاه كما يلي:

يقوم الحاسوب باختبار صحة الشرط *a > b* أي اختبار فيما إذا كان محتوى المتغيرة *a* أكبر من محتوى المتغيرة *b* حيث:

- إذا كان الشرط صحيحا أي إذا كان محتوى *a* أكبر من محتوى *b* فإن الحاسوب يقوم بتنفيذ التعليمة

writeln ('a Est Supérieure A b ')

- إذا كان الشرط غير صحيحا أي إذا كان محتوى *a* ليس أكبر (أصغر) من محتوى *b* فإن الحاسوب يقوم بتنفيذ التعليمة

writeln ('a Est Inferieure A b ') ;

و عليه يصبح البرنامج كما يلي:

```
program Instruction_IF ;
var  a,b  : integer ;
begin
      write (' DONNER LA VALEUR DE a : ');
      readln (a) ;
      write (' DONNER LA VALEUR DE b : ');
      readln (b) ;
      If a > b Then writeln ('a Est  Supérieure A b ')
```

```
Else writeln ('a Est  Inferieure A b ') ;
readln ;
end.
```

عند تنفيذ البرنامج نحصل على ما يلي:

النشر على الشاشة

```
DONNER LA VALEUR DE a :12
DONNER LA VALEUR DE b :8
a Est  Supérieure A b
```

عند تنفيذ البرنامج مرة ثانية نحصل على ما يلي:

النشر على الشاشة

```
DONNER LA VALEUR DE a :25
DONNER LA VALEUR DE b :40
a Est  Inferieure A b
```

مثـال 13.08:

يتم استخدام التعليمة الشرطية IF….THEN….ELSE…. للمثال 07.08 أعلاه كما يلي:

If *a = b* **Then** *writeln ('a Egale A b ')*

Else *writeln ('a Different De b ') ;*

يتم تنفيذ التعليمة IF….THEN….ELSE…. أعلاه كما يلي:

يقوم الحاسوب باختبار صحة الشرط $a = b$ أي اختبار فيما إذا كان محتوى a يساوي محتوى b حيث:

- إذا كان الشرط صحيحا أي إذا كان محتوى a يساوي محتوى b فإن الحاسوب يقوم بتنفيذ التعليمة

writeln ('a Egale A b ')

- إذا كان الشرط غير صحيحا أي إذا كان محتوى a لا يساوي (يختلف) من محتوى

b فإن الحاسوب يقوم بتنفيذ التعليمة

writeln (‘a Different De b ‘)

و عليه يصبح البرنامج كما يلي:

```
program Instruction_IF ;
var  a,b  : integer ;
begin
write (' DONNER LA VALEUR DE a : ');
readln (a) ;
write (' DONNER LA VALEUR DE b : ');
readln (b) ;
If a = b Then writeln ('a Egale A b ')
        Else writeln ('a Different De b ') ;
readln ;
end.
```

عند تنفيذ البرنامج نحصل على ما يلي:

النشر على الشاشة

```
DONNER LA VALEUR DE a :100
DONNER LA VALEUR DE b :578
a  Different  De  b
```

عند تنفيذ البرنامج مرة ثانية نحصل على ما يلي:

النشر على الشاشة

```
DONNER LA VALEUR DE a :123
DONNER LA VALEUR DE b :123
a  Egale  A  b
```

مثال 14.08:

يتم استخدام التعليمة الشرطية ....IF....THEN....ELSE للمثال 11.08 أعلاه كما يلي:

If *sal <= 50000* **Then** *imp:= 0.25*sal*

Else *imp:= 0.30*sal ;*

و عليه يصبح البرنامج كما يلي:

```
program Instruction_IF ;
var  sal  : integer ;
   imp : real ;
begin
      write (' DONNER LE SALAIRE : ');
      readln (sal) ;
      if sal <= 50000 Then imp:= 0.25*sal
                      Else imp:= 0.30*sal  ;
      writeln ('Les Impots = ', imp);
      readln ;
end.
```

ملاحظة 04.08:

من خلال كل من طريقة كتابة التعليمة الشرطية IF و الأمثلة أعلاه نلاحظ أن التعليمة IF تحتوي على كل من ***Then*** و ***Else*** . هناك طريقة كتابة أخرى للتعليمة IF أين تحتوي فقط على ***Then***، أي لا تحتوي على ***Else*** . طريقة الكتابة هذه هي التالية:

If Condition **Then** Instruction_1 ;

تبعا لطريقة الكتابة أعلاه فإنه يتم تنفيذ التعليمة Instruction_1 فقط في حالة تحقق الشرط Condition مثلما يوضحه المثال التالي:

مثال 15.08:

If a<0 **Then** write ('a est négative') ;

إذا كان هناك العديد من التعليمات يجب تنفيذها في حالة تحقق الشرط، فإن طريقة الكتابة هي التالية:

If Condition **Then** Begin

Instruction_1 ;

Instruction_2 ;

```
        Instruction_3 ;
        …… ;
        …… ;
      Instruction_n ;
    End;
```

مثال 16.08:

```
program Instruction_IF ;
var  a : integer ;
     r : real ;
begin
   write (' DONNER LA VALEUR DE a : ');
   read(a);
   If a<0 Then begin
                writeln ( 'a est négative' ) ;
                writeln ( 'imposible de calculer la racine') ;
                a := - a ;
                r :=sqrt(a) ;
                writeln ( 'la racine de -a égale : ',r) ;
                writeln ( 'MERCI') ;
               end ;
    writeln ( 'MERCI') ;
end.
```

بعد أن يتم إدخال قيمة المتغيرة a يتم تنفيذ التعليمة **If** أعلاه كما يلي:

- إذا كانت قيمة a أقل من الصفر، يقوم الحاسوب بتنفيذ كتلة التعليمات المحصورة بين **begin** و **end ;** ثم ينتقل إلى بعد ذلك إلى مواصلة تنفيذ البرنامج أي تنفيذ writeln ('MERCI') ثم end
- إذا لم تكن قيمة a أقل من الصفر، يقوم الحاسوب بمواصلة تنفيذ البرنامج أي تنفيذ writeln ('MERCI') ثم end

ملاحظة 05.08:

من خلال جميع الأمثلة أعلاه نلاحظ أن عدد التعليمات التي يتم تنفيذها سواء بعد ***Then*** أو

بعد ***Else*** يساوي تعليمة واحدة. يمكن تنفيذ أكثر من تعليمة واحدة (مجموعة من التعليمات) بعد ***Then*** أو بعد ***Else*** شريطة أن يتم تجميع هذه التعليمات فيما يسمى بكتلة أو مجموعة تعليمات Un Bloc D'instructions أو التعليمة المركبة Une Instruction Composée مثلما يوضحه المثال أدناه:

مثـال 17.08:

```
program Instruction_IF ;
var  salb  : integer ;
   imp , saln : real ;
begin
      write (' DONNER LE SALAIRE BRUT : ');
      readln (sabl) ;
      if salb <= 50000 Then
                        begin
                            imp := 0.25*sal ;
                           saln := salb – imp;
                        end
                        Else
                        begin
                            imp := 0.30*sal ;
                           saln := salb – imp;
                        end
      writeln ('SALAIRE  BRUT  = ', salb);
      writeln ('Les Impots = ', imp);
      writeln ('SALAIRE NET = ', saln);
      readln ;
end.
```

04.08 – كتلة تعليمات أو التعليمة المركبة:

Bloc D'instructions ou Instruction Composée :

كتلة تعليمات أو التعليمة المركبة عبارة عن مجموعة أو سلسلة من التعليمات المفصولة فيما بينها لفاصلة منقوطة و التي تبدأ بالكلمة المحجوزة **begin** Le Mot Réservé و تنتهي بالكلمة المحجوزة **end**

طريقة كتابة التعليمة المركبة هي التالية:

```
Begin
    Instruction_1 ;
    Instruction_2 ;
    Instruction_3 ;
    Instruction_4 ;
    …… ;
    …… ;
    …… ;
    Instruction_n ;
End
```

المثال أدناه يقدم تعليمتين مركبتين

مثـال 18.08:

- التعليمة المركبة الأولى:

```
Begin
   x := a + 5 ;
   j := 2 ;
   z := x + j ;
   Write (x) ;
end
```

- التعليمة المركبة الثانية:

```
Begin
   a :=u * 5 ;
   y := a + I  ;
end
```

05.08 – التعليمات IF المتداخلة: Les Instructions IF Imbriquées

هناك بعض المسائل التي تحتاج إلى إدماج تعليمتين IF….THEN….ELSE… (أو أكثر) داخل بعضهما البعض. طريقة كتابة التعليمات IF المتداخلة هي التالية:

```
If Condition_1 Then If Condition_2 Then Instruction_1
                                   Else  Instruction_2
                   Else  Instruction_3 ;
```

يتم تنفيذ التعليمات IF المتداخلة الموضحة من خلال طريقة الكتابة أعلاه كما يلي:

- يقوم الحاسوب باختبار الشرط الأول Condition_1 أي تنفيذ **IF** الأولى حيث:
  - إذا كان صحيحا يمر إلى اختبار الشرط الثاني Condition_2 أي يمر إلى **IF** الثانية
  - إذا لم يكن صحيحا يقوم بتنفيذ التعليمة Instruction_3 ليواصل بعد ذلك تنفيذ بقية تعليمات البرنامج
- عندما يكون الشرط الأول Condition_1 صحيحا، يمر الحاسوب إلى اختبار الشرط الثاني Condition_2 أي المرور إلى **IF** الثانية حيث:
  - إذا كان صحيحا يقوم بتنفيذ التعليمة Instruction_1 ليواصل بعد ذلك تنفيذ بقية تعليمات البرنامج
  - إذا لم يكن صحيحا يقوم بتنفيذ التعليمة Instruction_2 ليواصل بعد ذلك تنفيذ بقية تعليمات البرنامج

بغرض توضيح كيفية استخدام التعليمات IF المتداخلة، نأخذ الأمثلة التالية:

مثـال 19.08:

```
program Instruction_IF_IMBRIQEES ;
var  a,b,c,d : integer ;
begin
write (' DONNER LA VALEUR DE A : ');
read(a);
write (' DONNER LA VALEUR DE B : ');
read(b);
write (' DONNER LA VALEUR DE C : ');
read(c);
if a<b then if  b < c then writeln ( ' A < B < C ' )
                      else writeln ( ' A < B > C ' )
       else writeln ('A>=B  et  B  SUPER  OU  INFER  OU  EGALE  C ') ;
writeln ( 'M  E  R  C  I  ');
readln ;
end.
```

يتم تنفيذ البرنامج أعلاه بعد إدخال قيم المتغيرات a,b,c كما يلي:

- يقوم الحاسوب باختبار الشرط الأول ***a<b*** أي تنفيذ **IF** الأولى حيث:
  - إذا كان صحيحا يمر إلى اختبار الشرط الثاني ***b < c*** أي يمر إلى **IF** الثانية
  - إذا لم يكن صحيحا يقوم بتنفيذ التعليمة

writeln (' A >=B et B SUPER OU INFER OU EGALE C ')

ليواصل بعد ذلك تنفيذ بقية تعليمات البرنامج و هي *writeln ('M E R C I ')*

- عندما يكون الشرط الأول ***a<b*** صحيحا، يمر الحاسوب إلى اختبار الشرط الثاني ***b < c*** أي المرور إلى **IF** الثانية حيث:

– إذا كان صحيحا يقوم بتنفيذ التعليمة ***writeln (' A < B < C ')*** ليواصل بعد ذلك تنفيذ بقية تعليمات البرنامج *writeln ('M E R C I ')*

– إذا لم يكن صحيحا يقوم بتنفيذ التعليمة ***writeln (' A < B > C ')*** ليواصل بعد ذلك تنفيذ بقية تعليمات البرنامج *writeln ('M E R C I ')*

مثـال 20.08:

```
program Instruction_IF_IMBRIQEES ;
var  a,b,c,d : integer ;
begin
write (' DONNER LA VALEUR DE A : ');
read(a);
write (' DONNER LA VALEUR DE B : ');
read(b);
write (' DONNER LA VALEUR DE C : ');
read(c);
if a<b then if b<c then writeln(' A < B < C ')
                   else writeln(' A < B > C')
       else if b>c then writeln (' A > B > C')
                   else writeln (' A > B < C');
writeln ( 'M E R C I ');
readln ;
end.
```

مثـال 21.08:

```
if a<b then if b<c then writeln(' A < B < C ')
                   else writeln(' A < B > C')
       else if b>c then writeln (' A > B > C')
                   else writeln (' A > B < C');
```

مثـال 22.08:

```
if a<b then if b<c then writeln(' A < B < C ')
                   else if b>c then writeln(' A < B > C')
                                else writeln(' A < B = C')
```

```
else if  b>c then writeln (' A > B > C')
else writeln (' A > B < C');
```

مثال 23.08:

```
if a<b then if b<c then writeln(' A < B < C ')
else if b>c then writeln(' A < B > C')
else writeln(' A < B = C')
else if a=b then if b<c then writeln (' A = B < C')
else writeln(' A = B > C ')
else if b<c then writeln (' A > B < C')
else writeln(' A > B > C ');
```

06.08 – التوجيه المتعدد: التعليمة L'aiguillage Multiple CASE

ليكن البرنامج المثال التالي:

مثال 24.08:

```
program Instruction ;
var  a  : integer ;
begin
    write (' DONNER LA VALEUR DE LA VARIABLE ENTIERE  A : ');
    read(a);
    if a=1 then writeln (' UN ');
    if a=2 then writeln (' DEUX ');
    if a=3 then writeln (' TROIS ');
    if a=4 then writeln (' QUATRE ');
    writeln ( 'M  E  R  C  I  ');
    readln;
end.
```

التعليمات IF الأربعة يمكن تعويضها أو استبدالها بتعليمة واحدة هي التعليمة CASE كما يلي:

```
CASE a OF
  1 : writeln(' UN ' );
  2 : writeln(' DEUX ' );
```

```
        3 : writeln(' TROIS ' );
        4 : writeln(' QUATRE ' );
      END;
```

و عليه يصبح برنامج المثال 24.08 كما يلي:

```
program Instruction_CASE_OF ;
var  a  : integer ;
begin
   write (' DONNER LA VALEUR DE LA VARIABLE ENTIERE  A : ');
   read(a);
   CASE a OF
      1 : writeln(' UN ' );
      2 : writeln(' DEUX ' );
      3 : writeln(' TROIS ' );
      4 : writeln(' QUATRE ' );
   END;
   writeln ( 'M  E  R  C  I  ');
   readln;
end.
```

تبعا للمثال أعلاه، فإن طريقة الكتابة للتعليمة CASE هي التالية:

```
CASE  Expression OF
        Domaine_1 : Instruction_1 ;
        Domaine_2 : Instruction_2 ;
        Domaine_3 : Instruction_3 ;
        ……
        ……
        ……
        Domaine_n : Instruction_n ;
END ;
```

حيث أن:

Expression : تمثل متغيرة أو عملية حسابية

Domaine_1 : تمثل قيمة أو مجال من القيم

بغرض توضيح كيفية استخدام التعليمة CASE نأخذ الأمثلة التالية:

مثال 25.08:

المثال أدناه يوضح حالة Expression عبارة عن عملية حسابية

```
program Instruction_CASE_OF ;
var  a  : integer ;
begin
   write (' DONNER LA VALEUR DE LA VARIABLE ENTIERE  A : ');
   read(a);
   CASE a+5 OF
      10 : writeln(' DIX ' );
      20 : writeln(' VINGT ' );
      30 : writeln(' TRENTE ' );
      40 : writeln(' QUARANTE ' );
   END;
   writeln ( 'M  E  R  C  I  ');
   readln;
end.
```

مثال 26.08:

المثال أدناه يوضح حالة Domaine_1 عبارة عن مجال من القيم

```
program Instruction_CASE_OF ;
var  a  : integer ;
begin
   write (' DONNER LA VALEUR DE LA VARIABLE ENTIERE  A : ');
   read(a);
   CASE a OF
      10..15 : writeln(' DIX…….QUINZE ' );
      20..30 : writeln(' VINGT …….TRENTE ' );
      35..40 : writeln(' TRENTE CINQ…….QUARANTE ' );
      41..50 : writeln(' QUARANTE ET UN……CINQUANTE ' );
```

```
    END;
    writeln ( 'M  E  R  C  I  ');
    readln;
end.
```

كما هناك طريقة كتابة ثانية للتعليمة CASE هي التالية:

```
CASE  Expression OF
        Domaine_1 : Instruction_1 ;
        Domaine_2 : Instruction_2 ;
        Domaine_3 : Instruction_3 ;
        ……….
        ……….
        ……….
        Domaine_n : Instruction_n ;
ELSE
        Instruction_1 ;
        Instruction_2 ;
       ……..
       ……..
        Instruction_m ;
END ;
```

بغرض توضيح طريقة الكتابة الثانية نأخذ المثال التالي:

مثـال 27.08:

ليكن البرنامج التالي:

```
program Instruction_ CASE_OF;
var  a  : integer ;
begin
write (' DONNER LA VALEUR DE A COMPRISE ENTRE 1 ET 4 : ');
read(a);
CASE a OF
     1 : writeln(' UN ' );
```

```
        2 : writeln(' DEUX ' );
        3 : writeln(' TROIS ' );
        4 : writeln(' QUATRE ' );
    ELSE
      writeln (' LA VALEUR n''est PAS COMPRISE ENTRE 1 ET 4 ');
END;
writeln ( 'M  E  R  C  I  ');
readln ;
end.
```

تمارين الفصل

✍ التمرين 01.08:

1. ما هي تعليمة القفز اللاشرطية L'instruction De Branchement Inconditionnel

✍ التمرين 02.08:

1. قدم طريقة الكتابة لتعليمة القفز اللاشرطية المتوصل إليها في التمرين 01.08 أعلاه.

✍ التمرين 03.08:

2. ما هي تعليمة التي تسمح بالتصريح بالعلامات Les Etiquettes.

3. قدم طريقة الكتابة للتعليمة المستخدمة للتصريح بالعلامات Les Etiquettes.

4. قدم طريقة الكتابة للتصريح بالعديد من العلامات Les Etiquettes.

✍ التمرين 04.08:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program XXXX ;
Var  a, b, c ,d : integer ;
begin
    a:= 1500 ;
    b:= 5000 ;
    c:= a+b ;
    d:= b-a ;
    Writeln (a) ;
    Writeln (b) ;
    Writeln (c) ;
    Writeln (d) ;
end.
```

- **المطلوب:**

1. تبعا لمبدأ التسلسل في تنفيذ تعليمات أسطر البرنامج، بين بالتفصيل كيف يتم تنفيذ البرنامج أعلاه.

✍ التمـريـن 05.08:

ليكن برنامج التمرين 03.08 أعلاه

- **المطلـوب:**

1. باستخدام تعليمة القفز اللاشرطي ***goto*** قم بإدخال تعديلات على البرنامج بحث لا يتم تنفيذ السطر رقم 8 ; *Writeln (a)*

✍ التمـريـن 06.08:

ليكن برنامج التمرين 03.08 أعلاه

- **المطلـوب:**

1. باستخدام تعليمة القفز اللاشرطي ***goto*** قم بإدخال تعديلات على البرنامج بحث لا يتم تنفيذ الأسطر التالية:

- السطر رقم 8 ; *Writeln (a)*
- السطر رقم 9 ; *Writeln (b)*
- السطر رقم 10 ; *Writeln (c)*

✍ التمـريـن 07.08:

ليكن برنامج التمرين 03.08 أعلاه

- **المطلـوب:**

1. باستخدام تعليمة القفز اللاشرطي ***goto*** قم بإدخال تعديلات على البرنامج بحث لا يتم تنفيذ الأسطر التالية:

- السطر *a:= 1500 ;*
- السطر *b:= 5000 ;*
- السطر *c:= a+b ;*
- السطر *d:= b-a ;*

✍ التمـريـن 08.08:

ليكن برنامج التمرين 03.08 أعلاه

- **المطلـوب:**

1. باستخدام تعليمة القفز اللاشرطي ***goto*** قم بإدخال تعديلات على البرنامج بحث لا يتم تنفيذ الأسطر التالية:

- السطر *a:= 1500 ;*
- السطر *b:= 5000 ;*
- السطر *c:= a+b ;*
- السطر *d:= b-a ;*
- السطر *Writeln (a) ;*
- السطر *Writeln (b) ;*
- السطر *Writeln (c) ;*
- السطر *Writeln (d) ;*

✍ التمـريـن 09.08:

1. ما هي الأسطر في البرنامج التي لا يمكن الانتقال إليها بواسطة تعليمة القفز اللاشرطي ***goto***

✍ التمـريـن 10.08:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program ZZZ ;
var x,y,z : real ;
label 2 ;
begin
 writeln (' LA VILLE  DE  SOUGUEUR  EST  TRES  BELLE ' ) ;
Goto 2 ;
x:= 5.6 ;
y:= 1.2 ;
z:=x/y ;
Writeln (x) ;
Writeln (y) ;
Writeln (z) ;
2 : Writeln (' Merci De Votre Visite') ;
Writeln (' Merci De Votre Visite') ;
end.
```

- **المطلـوب:**

1. ما هي الأسطر (التعليمات) التي لا يقوم بتنفيذها الحاسوب في البرنامج أعلاه.

2. على ماذا نحصل على الشاشة عند تنفيذ البرنامج أعلاه.

✍ التمـريـن 11.08:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program ZZZ ;
var x,y,z : real ;
label 2 , 1 , 100 ;
begin
 writeln (' LA VILLE  DE  SOUGUEUR  EST  TRES  BELLE ' ) ;
Goto 2 ;
1 : x:= 5.6 ;
y:= 1.2 ;
z:=x/y ;
Writeln (x) ;
Writeln (y) ;
Writeln (z) ;
Goto 100 ;
2 : Writeln (' Merci De Votre Visite') ;
Goto 1 ;
100 : Writeln (' S O U G U E U R') ;
end.
```

- **المطلـوب:**

1. ما هي الأسطر (التعليمات) التي لا يقوم بتنفيذها الحاسوب في البرنامج أعلاه.

2. على ماذا نحصل على الشاشة عند تنفيذ البرنامج أعلاه.

✍ التمـريـن 12.08:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program yyyy ;
var x_1,y_1,z_1 : real ;
   x1,y1,z1,w1 : integer ;
   xx,yy,zz,ww : string ;
```

```
label 5 , 6 , 7 ;
begin
    GOTO 5 ;
    write ('Donnez La Valeur De x1 : ');
    readln (x1);
    write ('Donnez La Valeur De y1 : ');
    readln (y1);
    5: xx:='UNIVERSITE';
    yy:='IBN  KHALDOUN';
    zz:='  ' ;
    ww:= xx + zz + yy ;
    z1:= x1 + y1 ;
    w1:= x1 * y1 ;
    GOTO 6 ;
    write ('La Valeur De x1 : ');
    writeln (x1) ;
    write ('La Valeur De y1 : ');
    writeln (y1) ;
    6 : write ('La Valeur De WW Est : ');
    writeln ( ww );
    write ('La Valeur De Z1 Est : ');
    writeln ( z1 );
    goto 7 ;
    write ('La Valeur De W1 Est : ');
    writeln ( w1 );
7: end.
```

- **المطلـوب:**

1. بافتراض أننا نقوم بتنفيذ البرنامج أعلاه بدون التعليمات ***goto*** على ماذا نحصل.
2. ما هي الأسطر (التعليمات) التي لا يقوم بتنفيذها الحاسوب في البرنامج أعلاه.
3. على ماذا نحصل على الشاشة عند تنفيذ البرنامج أعلاه.

✍ التمـريـن 13.08:

1. قم بكتابة برنامج بلغة الـ PASCAL يحتوي على 5 تعليمات القفز اللاشرطي ***goto*** .

✍ التمـريـن 14.08:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program yyyy ;
var  x,y,z  : real ;
     x11,y11,z11 , w11  : integer ;
label  10 , 60   80
begin
        readln (x);
        readln (y);
        readln (x11);
        readln (y11);
        z11:= x11 + y11 ;
        w11:= x11 * y11 ;
        GOTO 80
        10   write ('La Valeur De x : ');
        writeln (x) ;
        write ('La Valeur De y : ');
        writeln (y) ;
        GOTO 60 ;
        80: write ('La Valeur De W11 Est : ');
        writeln ( w11 );
        write ('La Valeur De Z11 Est : ');
        writeln ( z11 );
        writeln ('M E R C I') ;
        GOTO 10
60  end.
```

- **المطلـوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.
2. قم بتصحيح الأخطاء المتوصل إليها في السؤال الأول أعلاه.
3. ماذا نحصل على الشاشة عند تنفيذ البرنامج أعلاه بعد تصحيحه.

✍ التمـريـن 15.08:

قم بكتابة برنامج بلغة الـ PASCAL يحتوي على 5 تعليمات ***goto***

✍ التمـريـن 16.08:

قدم مختلف طريق الكتابة للتعليمة الشرطية ....IF....THEN....ELSE

✍ التمـريـن 17.08:

نقوم بإدخال عن طريق لوحة المفاتيح معطيتين عدديتين (صحيحتين أو حقيقيتين) a و b حيث:

إذا كان $a > b$ نقوم بالنشر على الشاشة ما يلي:

a **Est Supérieure A** b

أما إذا لم يكن (أي إذا a يختلف عن b) نقوم بنشر على الشاشة ما يلي:

a **Est Inferieure A** b

- **المطلـوب:**

1. باستخدام التعليمة الشرطية ....IF…THEN….ELSE ، أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز ما هو أعلاه.

✍ التمـريـن 18.08:

نقوم بإدخال عن طريق لوحة المفاتيح معطيتين عدديتين (صحيحتين أو حقيقيتين) a و b حيث:

إذا كان $a = b$ نقوم بالنشر على الشاشة ما يلي:

a **Egale A** b

أما إذا لم يكن نقوم بنشر على الشاشة ما يلي:

a **Différent De** b

- **المطلـوب:**

1. باستخدام التعليمة الشرطية ....IF…THEN….ELSE ، أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز ما هو أعلاه.

✍ التمـريـن 19.08:

نقوم بإدخال عن طريق لوحة المفاتيح معطية عددية و لتكن x حيث:

إذا كان x >150 نقوم بالنشر على الشاشة ما يلي:

x **Est Supérieure A** 150

أما إذا لم يكن نقوم بنشر على الشاشة ما يلي:

x **Est Inferieure A** 150

- **المطلـوب:**

1. باستخدام التعليمة الشرطية ....IF…THEN….ELSE ، أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز ما هو أعلاه.

التمرين 20.08:

نقوم بإدخال عن طريق لوحة المفاتيح 3 علامات و لتكن $note_1, note_2, note_3$ ثم نقوم بحساب المعدل $moy = \frac{note_1 + note_2 + note_3}{3}$ حيث:

إذا كان المعدل أقل من 10 أي $moy >= 10$ نقوم بالنشر على الشاشة ما يلي:

L'étudiant Admis

أما إذا لم يكن نقوم بنشر على الشاشة ما يلي:

L'étudiant Ajourné

- **المطلوب:**

1. باستخدام التعليمة الشرطية ....IF....THEN....ELSE ، أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز ما هو أعلاه.

التمرين 21.08:

نقوم بإدخال عن طريق لوحة المفاتيح الأجر الذي يتلقاه موظف ما و الذي نسميه *sal* ثم نقوم بحساب مقدار الضريبة *imp* التي يدفعها هذا الموظف و التي تحسب كما يلي:

إذا كان الأجر يساوي أو يقل عن 50000 دج أي $sal <= 50000$ فإن مقدار الضريبة *imp* يساوي:

$$imp = 0.25 \times sal$$

أما إذا لم يكن فإن مقدار الضريبة *imp* يساوي:

$$imp = 0.30 \times sal$$

- **المطلوب:**

1. باستخدام التعليمة الشرطية ....IF....THEN....ELSE ، أكتب البرنامج بلغة الـ PASCAL الذي يسمح بانجاز ما هو أعلاه.

التمرين 22.08:

لتكن طرق الكتابة للتعليمة الشرطية ....IF....THEN....ELSE التالية:

1. الحالة الأولى:

```
IF  x < y  THEN y:= x + 125 ;
           ELSE  y:= x - 125
```

2. الحالة الثانية:

```
IF  x <> y  THEN x:= y * 215 ;
            ELSE  x:= y / 215 ;
```

3. الحالة الثالثة:

```
IF  x := 450  THEN x:= y * 215
              ELSE  x:= y / 215 ;
```

4. الحالة الرابعة:

```
IF   x > y   THEN x:= y * 215
             ELSE  x:= y / 215 ;
```

- **المطلوب:**

1. حدد الأخطاء في كل حالة من الحالات أعلاه ثم قم بتصحيحها.

✍ التمرين 23.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة x.
2. إدخال قيمة y.
3. إذا x > y **إذن** $z=x+y$

   و إلا $z=y-x$
4. نشر قيمة z

- **المطلوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمرين 24.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة x.
2. إدخال قيمة y.
3. إذا x = y **إذن** $z=x*y$

   و إلا $z=y/x$
4. نشر قيمة z

- **المطلوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمرين 25.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال العلامة الأولى *NOTE_1* .

2. إدخال العلامة الثانية $NOTE_2$.
3. إدخال العلامة الثالثة $NOTE_3$.
4. إدخال العلامة الرابعة $NOTE_4$.
5. حساب المعدل MOY
6. إذا $MOY <= 10$ **إذن** أنشر *Etudiant Admis*

و إلا أنشر *Etudiant Ajourné*

- **المطلوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 26.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة y .
2. إذا كان $y < 0$ **إذن**

أنشر *La Valeur De y est Négative*

$$y = y + 125$$

و إلا

أنشر *La Valeur De y est Négative*

$$y = y - 150$$

3. أنشر قيمة y

- **المطلوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 27.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة k .
2. إذا كان $0 \geq y \leq 100$ **إذن**

أنشر $La\ Valeur\ De\ y \in [0,100]$

$$y = y + 5$$

$$z = y + 1500$$

$$f = y * 0.005$$

و إلا

أنشر $La\ Valeur\ De\ y \notin [0,100]$

$$y = y - 5$$

$$z = y - 1500$$

$$f = y * 0.001$$

3. أنشر قيمة y
4. أنشر قيمة z
5. أنشر قيمة f

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 28.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة كل من vat_1 و vat_2.
2. إذا كان $val_1 > 100$ و $val_2 > 100$ **إذن**

$$f(x) = \ln\sqrt{val_1 \times val_2}$$

أنشر vat_1

أنشر vat_2

أنشر $f(x)$

و إلا

إعادة إدخال قيمة كل من vat_1 و vat_2

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 29.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة كل من n و m و l
2. إذا كان $n \geq 150$ و $m > 100$ **إذن**

$$r = \sqrt{\frac{n \times m}{5}}$$

و إلا

$$r = \left(\frac{n \times m}{5}\right)^2$$

3. أنشر كل من n و m و l و r

- **المطلوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 30.08:

باستخدام التعليمة الشرطية ....IF....THEN....ELSE أكتب البرنامج الموافق لما يلي:

1. إدخال قيمة z
2. حساب قيمة $g(z)$ حيث أن:

$$g(z) = \begin{cases} \left(\frac{z-120}{230}\right)^2 & 12 < z < 25 \\ \left(\frac{150+5z}{30}\right) & \textit{Alleurs} \end{cases}$$

3. نشر $g(z)$

✍ التمـريـن 31.08:

تقدم لك طريقة كتابة تعليمتين IF المتداخلتين كما يلي:

If Condition_1 **Then If** Condition_2 **Then** Instruction_1
Else Instruction_2
Else Instruction_3 ;

- **المطلوب:**

1. وضح بالتفصيل كيف يتم تنفيذ التعليمتين IF المتداخلتين.

✍ التمـريـن 32.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة z
2. حساب قيمة $g(z)$ حيث أن:

$$g(z)=\begin{cases}\left(\dfrac{z-120}{230}\right)^2 & 12<z<25\\ \left(\dfrac{150+5z}{30}\right) & 25<z<50\\ 0 & \textit{Alleurs}\end{cases}$$

3. نشر $g(z)$

- **المطلـوب:**

1. وضح بالتفصيل كيف يتم تنفيذ ال تعليمتين IF المتداخلتين.

✍ التمـريـن 33.08:

ليكن البرنامج بلغة الـ PASCAL التالي:

```
program Instruction_IF_IMBRIQEES ;
var  a,b,c,d : integer ;
begin
write (' DONNER LA VALEUR DE A : ');
read(a);
write (' DONNER LA VALEUR DE B : ');
read(b);
write (' DONNER LA VALEUR DE C : ');
read(c);
if a<b then if  b < c then writeln ( ' A < B < C ' )
                      else writeln ( ' A < B > C ' )
       else writeln ('A>=B et B SUPER OU INFER OU EGALE C ') ;
writeln ( 'M  E  R  C  I  ');
readln ;
end.
```

- **المطلـوب:**

1. وضح بالتفصيل ماذا يحدث عند تنفيذ البرنامج أعلاه.

✍ التمـريـن 34.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال العلامة الأولى $NOTE_1$.
2. إدخال العلامة الثانية $NOTE_2$.

3. إدخال العلامة الثالثة $NOTE_3$.
4. إدخال العلامة الرابعة $NOTE_4$.
5. حساب المعدل MOY
6. **إذا** $MOY \geq 10$ **إذن** أنشر $Etudiant \quad Admis$

و إلا إذا $MOY \geq 8$ **إذن** أنشر $Etudiant \quad Ajourné$

و إلا أنشر $Etudiant \quad Exclu$

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 35.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة كل من a و b
2. **إذا** $a > b$ **إذن** أنشر $a \quad est \quad plus \quad grand$

و إلا إذا $a < b$ **إذن** أنشر $b \quad est \quad plus \quad grand$

و إلا أنشر $a \quad = \quad b$

3. أنشر قيمة a
4. أنشر قيمة b
5. أنشر $MERCI$

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 36.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة x محصورة بين 1 و -1 أي $x \in]-1,+1[$
2. **إذا** $x > 1$ أو $x < -1$ **إذن** إعادة إدخال قيمة x

و إلا

$$y = \ln(x)$$

نشر قيمة x

نشر قيمة y

- **المطلوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمرين 37.08:

لتكن المعادلة من الدرجة الأولى $ax+b=c$ بغرض حل الجملة نتبع الخطوات التالية:

1. إدخال قيمة a حيث أن $a \neq 0$
2. **إذا** $a=0$ **إذن** إعادة إدخال قيمة a
3. إدخال قيمة b حيث أن $b \neq 0$
4. **إذا** $b=0$ **إذن** إعادة إدخال قيمة b
5. إدخال قيمة c حيث أن $c \neq 0$
6. **إذا** $c=0$ **إذن** إعادة إدخال قيمة c
7. حساب قيمة x حيث أن $x=\frac{c-b}{a}$
6. نشر قيمة x

- **المطلوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمرين 38.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة a حيث أن $a \neq 0$
2. **إذا** $a=0$ **إذن** إعادة إدخال قيمة a
3. إدخال قيمة b حيث أن $b \neq 0$
4. **إذا** $b=0$ **إذن** إعادة إدخال قيمة b
5. إدخال قيمة c حيث أن $c \neq 0$
6. **إذا** $c=0$ **إذن** إعادة إدخال قيمة c
7. حساب قيمة المحدد $delta$ حيث أن $delta=b^2-4ac$
8. **إذا** $delta=0$ **إذن**

حساب $x_1=\frac{-b}{2a}$

نشر *on a une solution multiple*

نشر x_1

و إلا إذا $delta < 0$ **إذن** أنشر *Pas De Solution*

و إلا

حساب $x_1 = \frac{-b - \sqrt{delta}}{2a}$

حساب $x_2 = \frac{-b + \sqrt{delta}}{2a}$

نشر *on a deux solutions*

نشر x_1

نشر x_2

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 39.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة x
2. إدخال قيمة y
3. حساب قيمة $g(x,y)$ حيث أن:

$$g(x,y) = \begin{cases} x + \frac{y}{5} - 5xy & x > 0, y \neq 0 \\ 78x + \frac{5}{2y + 4x} & x < 0, y = 0 \end{cases}$$

4. نشر $g(x,y)$

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 40.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال علامات خمسة مقاييس

2. حساب المعدل *moyenne*
3. **إذا** $moyenne \geq 16$ **إذن** أنشر *Félicitati on*
4. **إذا** $13 \leq moyenne < 16$ **إذن** أنشر *Trés Bien*
5. **إذا** $10 \leq moyenne < 13$ **إذن** أنشر *Bien*
6. **و إلا** أنشر *Exclu*

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 41.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة المتغيرة من النوع حرف *ch*
2. **إذا** $ch = 'a'$ أو $ch = 'A'$ **إذن** نشر *ch*

و إلا إعادة إدخال المعطية من النوع حرف

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 42.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة المتغيرة من النوع سلسلة أحرف *chaine*
2. **إذا** $chaine = 'zahra'$ أو $chaine = 'ZAHRA'$ **إذن** نشر *chaine*

و إلا إعادة إدخال المعطية من النوع سلسلة أحرف

- **المطلـوب:**

1. أكتب البرنامج الموافق للخطوات أعلاه.

✍ التمـريـن 43.08:

ليكن البرنامج التالي:

```
program Instruction_IF ;
var  x , y : integer ;
begin
    write ('donnez la valeur de x : ');
    read(x);
```

```
    write ('donnez la valeur de y : ');
    read(y);
    if x<y then write ('x est inférieur a y ')
    readln ;
end.
```

- **المطلوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.

2. قم بتصحيح الأخطاء المتوصل إليها في المطلوب الأول.

✍ التمـريـن 44.08:

ليكن البرنامج التالي:

```
program Instruction_IF ;
var  x , y : integer ;
begin
    write ('donnez la valeur de x : ');
    read(x);
    write ('donnez la valeur de y : ');
    read(y);
    if x<y then write ('x est inférieur a y ');
        else write ('x est superieure a y ')
    readln ;
end.
```

- **المطلوب:**

1. حدد الأخطاء التي يحتويها البرنامج أعلاه.

2. قم بتصحيح الأخطاء المتوصل إليها في المطلوب الأول.

✍ التمـريـن 45.08:

قدم مختلف طرق كتابة تعليمة التوجيه المتعدد CASE

✍ التمـريـن 46.08:

لتكن الخطوات التالية لإنجاز برنامج بلغة الـ PASCAL

1. إدخال قيمة المتغيرة الصحيحة e
2. إذا $e=3$ أنشر A
3. إذا $e=5$ أنشر B

4. إذا $e = 10$ أنشر D

✍ التمـريـن 47.08:

ليكن البرنامج التالي:

```
program Instruction_CASE_OF ;
var  a  : integer ;
begin
   write (' DONNER LA VALEUR DE LA VARIABLE ENTIERE  A : ');
   read(a);
   CASE a OF
            10..15 : writeln(' DIX…….QUINZE ' );
            16..30 : writeln(' SEIZE …….TRENTE ' );
            31..40 : writeln(' TRENTE CINQ…….QUARANTE ' );
            41..50 : writeln(' QUARANTE ET UN……CINQUANTE ' );
     ELSE
       writeln (' LA VALEUR n''est PAS COMPRISE ENTRE 10 ET 50');
   END;
   writeln ( 'M  E  R  C  I  ');
   readln;
end.
```

- **المطلـوب:**

1. ماذا يقوم البرنامج عند إدخال القيم التالية:

- 10
- 15
- 13
- 30
- 41
- 48
- 2
- 51
- 100
- 245

✍ التمـريـن 48.08:

ليكن البرنامج التالي:

```
program Instruction_ CASE_OF;
var  a  : integer ;
begin
write (' DONNER LA VALEUR DE A COMPRISE ENTRE 1 ET 4 : ');
read(a);
CASE a OF
      1 : writeln(' UN ' );
      2 : writeln(' DEUX ' );
      3 : writeln(' TROIS ' );
      4 : writeln(' QUATRE ' );
   ELSE
     writeln (' LA VALEUR n''est PAS COMPRISE ENTRE 1 ET 4 ');
END;
writeln ( 'M  E  R  C  I  ');
readln ;
end.
```

- **المطلـوب:**

1. ماذا يقوم البرنامج عند إدخال القيم التالية:

- 1
- 3
- 2
- 4
- 5
- 0
- 34
- 12

✍ التمـريـن 49.08:

ليكن البرنامج التالي:

```
program Instruction_CASE_OF ;
```

```
var  a  : integer ;
begin
write (' DONNER LA VALEUR DE LA VARIABLE ENTIERE  A : ');
read(a);
CASE a OF
   10..15 : writeln(' DIX…….QUINZE ' );
   20..30 : writeln(' VINGT …….TRENTE ' );
   35..40 : writeln(' TRENTE CINQ…….QUARANTE ' );
   41..50 : writeln(' QUARANTE ET UN……CINQUANTE ' );
END;
writeln ( 'M  E  R  C  I  ');
readln;
end.
```

- **المطلـوب:**

1. ماذا يقوم البرنامج عند إدخال القيم التالية:

- 15
- 12
- 6
- 10
- 51
- 46

✍ التمـريـن 50.08:

ليكن البرنامج التالي:

```
program Instruction_CASE_OF ;
var  a  : integer ;
begin
write (' DONNER LA VALEUR DE LA VARIABLE ENTIERE  A : ');
read(a);
CASE a OF
   1..5,10..12,20..30 : writeln(' PREMIER  CHOIX ' );
   31..40 : writeln(' DEUXIEMME  CHOIX ' );
   41..50 : writeln(' TROISIEMME  CHOIX ' );
   ELSE writeln ('MAUVAIS CHOIX');
```

```
END;
writeln ( 'M  E  R  C  I  ');
read(a);
end.
```

- **المطلوب:**

1. ماذا يقوم البرنامج عند إدخال القيم التالية:

- 6
- 45
- 5
- 2
- 55
- 21
- 44
- 33

الفصل 📖 التاسع

المعالجة التكرارية - الحلقات -

Le Traitement Répétitif - Les Boucles -

- La Boucle Avec Compteur
- La Boucle ***For … to … do …***
- La Boucle ***Repeat … until …***
- La Boucle ***While … do …***

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- المعالجة التكرارية.
- الحلقة.
- الحلقة بعداد.
- الحلقة ... *For ... to ... do* .
- الحلقة ... *While ... do* .
- الحلقة ... *Repeat ... until* .
- الحلقة ... *For ... downto ... do* .

01.09 – تمهيد:

في الكثير من المسائل نود القيام بإجراء عملية حسابية العديد من المرات، نذكر على سبيل المثال ما يلي:

1. نشر على الشاشة تعليق معين و ليكن MERCI عدد من المرات يساوي وليكن 150

في هذه الحالة البرنامج الموافق لذلك يحتوي على 150 سطر أو تعليمة التالية:

```
writeln('MERCI') ;
```

2. نشر على الشاشة 80 مرة ما يلي:

La Ville
De
Tiaret

في هذه الحالة البرنامج الموافق لذلك يحتوي على $80 \times 3 = 240$ سطر أو تعليمة التالية:

```
writeln('La Ville') ;
writeln('De') ;
writeln('Tiaret') ;
```

3. حساب و نشر مربع قيم y التالية:

$$y = -7,-6,-5,-4,-3,-2,-1,0,1,3,4,5,6,7,8,9,10,11,12,.....,85$$

في هذه الحالة نقوم بحساب و نشر مربع قيمة y حيث أن y يتغير بقيمة ثابتة تساوي الواحد من القيمة الابتدائية -7 إلى القيمة النهائية 85. أي حساب و نشر مربع قيمة y 93، البرنامج الموافق لذلك يحتوي على $93 \times 2 = 186$ سطر أو تعليمة التالية:

```
y :=sqr(-7) ;
writeln('y') ;
……….
……….
y :=sqr(85) ;
writeln('y') ;
```

4. حساب قيمة $z = 10X^2 + 5X - 15$ من أجل $X = 1,2,3,...,100$

في هذه الحالة نقوم بحساب قيمة z من أجل كل قيمة لـ X حيث أن X يتغير بقيمة ثابتة تساوي الواحد من القيمة الابتدائية 1 إلى القيمة النهائية 100. أي حساب قيمة z مائة مرة، البرنامج الموافق لذلك هو التالي:

```
z := 10(1)^2 + 5(1) – 15 ;
z := 10(2)^2 + 5(2) – 15 ;
z := 10(3)^2 + 5(3) – 15 ;
```

……
……
……
$z := 10(100)^2 + 5(100) - 15;$

البرنامج أعلاه سوف يحتوي على مائة تعليمة أو سطر.

من أجل تجنب كثرة الأسطر أو التعليمات سوف يتم استخدام ما يسمى بالحلقة (La Boucle) التي تشكل موضوع هذا الفصل.

02.09 – المعالجة التكرارية: الحلقة Le Traitement Répétitif : La Boucle

المعالجة التكرارية هي تلك المعالجة التي تتكرر عدد من المرات، هذه المعالجة التكرارية تسمى حلقة (Une Boucle). يتم الحصول على الحلقة عندما يتم تنفيذ تعليمة أو مجموعة من التعليمات عدد من المرات. بغرض التوضيح أكثر نأخذ الأمثلة التالية:

مثـال 01.09:

نريد تنفيذ الثلاث تعليمات التالية عدد من المرات

```
writeln('Bonne') ;
writeln('Journée') ;
writeln('Monsieur') ;
```

بترك التعليمات الثلاث أعلاه على حالها دون إضافة أي تعليمة أخرى، فإن الحاسوب يقوم بتنفيذها مرة واحدة فقط و بالتالي لم يتم الحصول على المعالجة التكرارية و من ثم لم يتم الحصول على الحلقة. بغرض الحصول على المعالجة التكرارية و من ثم الحصول على الحلقة يجب تنفيذ التعليمات الثلاثة أعلاه عدد من المرات دون إعادة كتابتها في البرنامج. يتم ذلك بالاستعانة أو استخدام تعليمة القفز اللاشرطية ***goto*** كما يلي:

```
10 : writeln('Bonne') ;
     writeln('Journée') ;
     writeln('Monsieur') ;
     goto 10 ;
```

بعد إضافة تعليمة القفز *goto* فإن الحاسوب يقوم بتنفيذ السطر الأول الممثل في التعليمة *writeln('Bonne') ;* ثم يتنقل إلى تنفيذ السطر الثاني الممثل في التعليمة *writeln('Journée') ;* ثم

يتنقل إلى تنفيذ السطر الثالث الممثل في التعليمة ; *writeln('Monsieur')* ثم يتنقل إلى تنفيذ السطر الرابع الممثل في التعليمة ; *goto 10* و التي تعيد الحاسوب إلى تنفيذ السطر الأول مرة أخرى و هكذا و عليه يصبح البرنامج الموافق لذالك هو التالي:

```
program La_BOUCLE_FOR ;
label 10 ;
begin
10 :writeln('Bonne') ;
writeln('Journée') ;
writeln('Monsieur') ;
goto 10 ;
end.
```

إن إضافة تعليمة القفز *goto* مكنت الحاسوب من إعادة تنفيذ التعليمات الثلاثة عدد من المرات و بالتالي تم الحصول على المعالجة التكرارية و من ثم تم الحصول على الحلقة.

ملاحظـة 01.09 :

- الحلقة تسمح بتنفيذ تعليمة أو مجموعة من التعليمات عدد من المرات، غير أن ما يعاب على الحلقة التي تم التطرق إليها من خلال المثال أعلاه أنها تسمح بتنفيذ تعليمة أو مجموعة من التعليمات عدد غير محدود أو غير منتهي من المرات.
- بغرض جعل الحلقة تقوم بتنفيذ تعليمة أو مجموعة من التعليمات عدد منتهي من المرات تم إجراء تعديلات على هذه الحلقات كما تم اقتراح حلقات أخرى بديلة. هذه التعديلات تتمثل في إدراج عداد Compteur ضمن هذه الحلقات و بذلك تم الحصول على الحلقة بعداد Boucle Avec Compteur

في لغة الـ Pascal لدينا أربعة أنواع من الحلقات هي:

1. الحلقة بعداد.
2. الحلقة ***For … To … Do…*** .
3. الحلقة ***Repeat … until …*** .
4. الحلقة ***While … do …*** .

سوف يتم التطرق بالتفصيل إلى كل نوع من خلال فقرات هذا الفصل

03.09 – الحلقة بعداد La Boucle Avec Compteur:

بافتراض أننا نود نشر على الشاشة 100 مرة ما يلي:

*

ZAHRA

البرنامج الذي يسمح بذلك هو التالي:

```
program BOUCLES ;
begin
        writeln('*');
        writeln('ZAHRA');
        writeln('*');
        writeln('ZAHRA');
        writeln('*');
        writeln('ZAHRA');
        …….. ;
        …….. ;
        …….. ;
        writeln('*');
        writeln('ZAHRA');
end.
```

البرنامج أعلاه يحتوي على 200 سطر أي الأسطر التالية:

```
        writeln('*');
        writeln('ZAHRA');
```

يتكرر كل منها 100 مرة

يمكن إجراء تعديل على البرنامج أعلاه بحيث ننقص أو نقلص في عدد أسطر البرنامج كما يلي:

```
program BOUCLES ;
label 10 ;
begin
        10: writeln('*');
        writeln('ZAHRA');
        goto 10 ;
end.
```

في هذه الحالة البرنامج أعلاه يقوم بنشر ما هو مطلوب عدد لا نهائي من المرات دون توقف و هذا بسبب وجود التعليمة ***goto 10*** التي في كل مرة تعيد الحاسوب إلى تنفيذ تعليمتي النشر *writeln('*')* و *writeln('ZAHRA')* .

إن وجود التعليمة *goto 10* سمح بتشكيل حلقة تتمثل في ما يلي:

```
10: writeln('*');
writeln('ZAHRA');
goto 10 ;
```

ما يقال عن هذه الحلقة أنها تسمح بنشر كل من * و *ZAHRA* عدد لا نهائي أو غير محدود من المرات و بغرض جعلها تقوم بعملية النشر بعدد محدود من المرات نقوم بإضافة **عداد** (Compteur) إلى هذه الحلقة، يتمثل دور هذا العداد في عد عدد مرات عمليات النشر التي تقوم بها الحلقة. هذا العداد يتكون من متغيرة و لتكن تساوي :

عبارة عن متغيرة و لتكن i تأخذ قيمة ابتدائية تساوي الصفر أي $i=0$ قبل بداية الحلقة، حيث يتم الرفع في قيمة المتغيرة i بالقيمة واحد بعد كل عملية نشر أي $i:=i+1$. بما أن $i:=i+1$ تقوم بعد عدد مرات أو عمليات النشر، فإنه سوف يتم إدراجها و وضعها مباشرة بعد التعليمتين المتعلقتين بالنشر. بعد إضافة العداد المكون من $i=0$ و $i:=i+1$ إلى الحلقة تصبح الحلقة كما يلي:

```
i :=0 ;
10: writeln('*');
writeln('ZAHRA');
i:=i+1 ;
goto 10 ;
```

ما يقال عن الحلقة الأخيرة أعلاه و المزودة بعداد أنها كذالك تسمح بنشر كل من * و *ZAHRA* عدد لا نهائي أو غير محدود من المرات و بغرض جعلها تقوم بعملية النشر عدد من المرات يساوي 100 فقط أي $i=100$ نقوم بإضافة إلى الحلقة الأخيرة تعليمة القفز الشرطية IF….THEN….ELSE كما يلي:

```
i :=0 ;
10: writeln('*');
writeln('ZAHRA');
i:=i+1 ;

if i=10 then goto 20 ;
goto 10 ;

20: writeln ('M E R C I ') ;

writeln ('M E R C I ') ;

readln ;

end.
```

- الشرط في هذه الحالة يتمثل في عدد مرات النشر يساوي القيمة 100 و المعبر عنه بـ: $i=100$ حيث:

1. إذا كان هذا الشرط غير محقق يقوم الحاسوب بمواصلة تنفيذ تعليمات البرنامج و هي:

goto 10 ;

التعليمة *goto 10* تعيد الحاسوب إلى الحلقة و التي تقوم بعملية نشر كل من * و *ZAHRA* إلى غاية تحقق الشرط (نشر 100 مرة) و من ثم الخروج من الحلقة

2. إذا كان هذا الشرط **محقق** يقوم الحاسوب بالخروج من الحلقة بواسطة التعليمة *goto 20* حيث يواصل تنفيذ بقية تعليمات البرنامج التالية:

```
20: writeln ('M E R C I ') ;
writeln ('M E R C I ') ;
readln ;
end.
```

و عليه يصبح البرنامج الذي يسمح بنشر كل من * و *ZAHRA* مائة مرة هو التالي:

```
program BOUCLES_AVEC_COMPTEUR ;
var i : integer ;
label 10 , 20 ;
begin
i:=0 ;
10: writeln('*');
writeln('ZAHRA');
i:=i+1;
if i=10 then goto 20 ;
goto 10 ;
20:writeln('M E R C I ') ;
writeln ('M E R C I ') ;
readln ;
end.
```

يمكن إدخال بعض التعديلات على البرنامج أعلاه غير أنه يبقى يحافظ على نفس الدور، هذه التعديلات تتمثل في:

- إضافة ***else*** إلى التعليمة **IF**
- تغيير موضع التعليمة *goto 10*

و عليه يصبح البرنامج كما يلي:

```
program BOUCLES_AVEC_COMPTEUR ;
var i : integer ;
label 10 , 20 ;
begin
i:=0 ;
10: writeln('*');
writeln('ZAHRA');
i:=i+1;
if i=10 then goto 20 else goto 10 ;
20:writeln('M E R C I ') ;
writeln ('M E R C I ') ;
readln ;
end.
```

يمكن إجراء تعديل على البرنامج أعلاه عن طريق تغيير الشرط كما يلي:

```
if i<100 then goto 10 else goto 20 ;
```

يمكن إجراء تعديل على البرنامج أعلاه كما يلي:

```
program BOUCLES_AVEC_COMPTEUR ;
var i : integer ;
label 10 ;
begin
i:=0 ;
10: writeln('*');
writeln('ZAHRA');
i:=i+1;
if i<10 then goto 10  ;
writeln('M E R C I ') ;
writeln ('M E R C I ') ;
readln ;
end.
```

من خلال المثال أعلاه يمكن أن نشير إلى أن الحلقة بعداد تتكون من:

- عداد Un Compteur: يتمثل في المتغيرة i التي تقوم بعد عدد مرات المعالجة (النشر) تبعا للتعليمة $i:=i+1$ هذه المتغيرة تأخذ قيمة ابتدائية تساوي 1 أي $i:=0+1=1$ و قيمة نهائية تساوي 10
- المعالجة Le Traitement : يتمثل في التعليمتين التاليتين:

writeln('');*

writeln('ZAHRA');

- الخطوة Le Pas : و التي تمثل مقدار زيادة أو رفع قيمة العداد أي مقدار زيادة قيمة المتغيرة i أي $i:=i+1$ في هذا المثال مقدار الزيادة (الخطوة) يساوي 1
- اختبار قيمة العداد Test : يتمثل في التعليمة ***if*** *i=10* ***then*** *goto 20* ***else*** *goto 10*

مثـال 02.09:

نريد حساب قيمة $z=10X^2+5X-15$ من أجل $X=0,1,2,3,...,150$ حيث يتم نشر قيمة z من أجل كل قيمة من قيم X.

البرنامج الذي يسمح بحساب و نشر قيمة z من أجل قيمة واحدة فقط للمتغير X يتكون من سطرين هما:

$z:=10*sqr(x)+5*x-15;$

Writeln (z) ;

أما البرنامج الذي يسمح بحساب و نشر قيمة z من أجل 151 قيمة للمتغير X يتكون من 302 سطر.

بغرض تجاوز كثرة الأسطر نقوم باستخدام الحلقة بعداد كما يلي:

x :=0 ;
10:
$z:=10*sqr(x)+5*x-15;$
Writeln (z) ;
x:=x+1;
goto 10 ;

ما يقال عن الحلقة أنها تقوم بعملية حساب و نشر قيمة z من أجل عدد لا نهائي من القيم للمتغير X و بغرض القيام بالعملية 151 فقط أي $X=151$ نقوم بالاستعانة بتعليمة القفز الشرطية IF….THEN….ELSE التي يتم إضافتها إلى الحلقة كما يلي:

x :=0 ;
10:
$z:=10*sqr(x)+5*x-15;$

```
Writeln (z) ;
x:=x+1;
if x=151 then goto 70 ;
goto 10 ;
70: end.
```

و عليه يصبح البرنامج كما يلي:

```
program BOUCLES_AVEC_COMPTEUR ;
var x , z : integer ;
label 10 , 70 ;
begin
x :=0 ;
10:
 z:=10*sqr(x)+5*x - 15 ;
Writeln (z) ;
x:=x+1;
if x=151 then goto 70 ;
goto 10 ;
70: readln ;
end.
```

نشير إلى أنه يمكن إجراء تعديلات على البرنامج أعلاه في الجزء الخاص بتعليمة القفز الشرطية IF

- *if <151 then goto 10 ;*
- *if <151 then goto 10 else goto 70 ;*

من خلال المثال أعلاه لدينا ما يلي:

- العداد : هو يتمثل في المتغيرة x التي تقوم بعد عدد مرات المعالجة (حساب + نشر) تبعا للتعليمة $x:=x+1$ هذه المتغيرة تأخذ قيمة ابتدائية تساوي 1 أي $x:=0+1=1$ و قيمة نهائية تساوي 151
- المعالجة Le Traitement : يتمثل في التعليمتين التاليتين:

```
z:=10*sqr(x)+5*x - 15 ;
Writeln (z) ;
```

- الخطوة Le Pas : و التي تمثل مقدار زيادة أو رفع قيمة العداد أي مقدار زيادة قيمة المتغيرة x أي $x:=x+1$ في هذا المثال مقدار الزيادة (الخطوة) يساوي 1
- اختبار قيمة العداد Test : يتمثل في التعليمة *if x=151 then goto 70*

مثـال 03.09:

نريد حساب قيمة $y = 15X + 150$ من أجل $X = 10,9,8,7,...,1,0,-1,-2,-3,-4,-5$ حيث يتم نشر قيمة y من أجل كل قيمة من قيم X.

البرنامج الذي يسمح بحساب و نشر قيمة y من أجل قيمة واحدة فقط للمتغير X يتكون من سطرين هما:

```
y := 15 * x + 150 ;
Writeln (y) ;
```

أما البرنامج الذي يسمح بحساب و نشر قيمة y من أجل 16 قيمة للمتغير X يتكون من $16 \times 2 = 32$ سطر. بغرض تجاوز كثرة الأسطر نقوم باستخدام الحلقة بعداد كما يلي:

```
x :=11 ;
10:
x:=x-1;
y := 15 * x + 150 ;
Writeln (y) ;
goto 10 ;
```

ما يقال عن الحلقة أنها تقوم بعملية حساب و نشر قيمة y من أجل عدد لا نهائي من القيم للمتغير X و بغرض القيام بالعملية 16 فقط نقوم بالاستعانة بتعليمة القفز الشرطية IF….THEN….ELSE التي يتم إضافتها إلى الحلقة كما يلي:

```
x :=10 ;
10:
x:=x-1;
y := 15 * x + 150 ;
Writeln (y) ;
if x=-5 then goto 70 ;
goto 10 ;
70: end.
```

و عليه يصبح البرنامج كما يلي:

```
program BOUCLES_AVEC_COMPTEUR ;
var x , y : integer ;
label 10 , 70 ;
begin
x :=11 ;
```

```
10:
x:=x-1;
y:=15*x+150 ;
Writeln ('X=  ',x,'  y= ',y) ;
if x=-5 then goto 70 ;
goto 10 ;
70: readln ;
end.
```

من خلال المثال أعلاه لدينا ما يلي:

- الخطوة Le Pas : و التي تمثل في هذا المثال مقدار نقصان و ليس زيادة قيمة العداد أي مقدار نقصان قيمة المتغيرة x أي $x:=x-1$ في هذا المثال مقدار النقصان (الخطوة) يساوي 1 أو نقول مقدار التغير يساوي -1
- القيمة الابتدائية للعداد تساوي 10
- القيمة النهائية للعداد تساوي -5
- القيمة الابتدائية للعداد أكبر من القيمة النهائية لذالك يكون مقدار الخطوة سالب
- اختبار قيمة العداد Test : يتمثل في التعليمة *if x=-5 then goto 70*

من خلال الأمثلة أعلاه يتضح أن الحلقة بعداد تتضمن ما يلي:

1. قيمة ابتدائية للعداد
2. قيمة نهائية للعداد
3. المعالجة Le Traitement
4. زيادة أو رفع (إضافة أ) قيمة العداد Incrémentation.
5. اختبار القيمة النهائية للعداد.

04.09 – الخطوة Le Pas:

مقدار تغير (زيادة أو نقصان) قيمة العداد يسمى بالخطوة. من خلال أمثلة الفقرة السابقة كانت الخطوة تساوي 1 و المعبر عنه بالتعليمتين $i:=i+1$ و $x:=x+1$ و $x:=x-1$. يمكن للخطوة أن تختلف (لا تساوي) عن 1 أي تكون تساوي قيمة أقل من 1 أو أكبر من 1 مثلما توضحه الأمثلة أدناه

مثـال 04.09:

نريد حساب نتائج العمليات التالية:

1. $S=1+1.5+2+2.5+3+3.5+4+4.5+.....+100+100.5$

2. $y = 200x + 15$ حيث أن $x = 5,10,15,20,25,30,35,.....,95,100,105$

1. العملية رقم 01: الخطوة تساوي 0.5 و عليه يصبح البرنامج هو التالي:

```
program BOUCLES_AVEC_COMPTEUR ;
var i , s : real ;
label 10 , 20 ;
begin
i:=1 ;
10:
s:=s+i ;
i:=i+0.5 ;
if i>100.5 then goto 20 else goto 10 ;
20 :
writeln (' S = ',s);
writeln ('MERCI');
readln ;
end.
```

2. العملية رقم 02: الخطوة تساوي 5 و عليه يصبح البرنامج هو التالي:

```
program BOUCLES_AVEC_COMPTEUR ;
var x , y : integer ;
label 10 , 20 ;
begin
x:=5 ;
10:
y:=200*x+15 ;
writeln ('x = ',x,' y = ',y) ;
x:=x+5 ;
if x>105 then goto 20 else goto 10 ;
20 :
writeln ('MERCI');
readln ;
end.
```

ملاحظـة 02.09 :

- إذا كانت القيمة الابتدائية للعداد **أقل** من القيمة النهائية، فإن الخطوة تكون موجبة
- إذا كانت القيمة الابتدائية للعداد **أكبر** من القيمة النهائية، فإن الخطوة تكون سالبة

05.09 – الحلقة ... *For ... To ... Do*... La Boucle:

تبعا لما تم التطرق إليه من خلال الفقرة الأخيرة أعلاه يتضح أن عملية الخروج من الحلقة بعداد (La Boucle Avec Compteur) تتم باستخدام أو بواسطة اختيار (Test) قيمة العداد بعد كل عملية زيادة أو إضافة، عملية الاختبار هذه تتم بواسطة التعليمة ....IF....THEN ، بحيث إذا وصلت قيمة العداد قيمة معينة و التي تسمى القيمة النهائية للعداد، فإنه يتم الخروج من الحلقة و إذا لم تصل قيمة العداد القيمة النهائية فإن الحاسوب يواصل تنفيذ محتوى الحلقة التي تتضمن مجموعة من التعليمات.

إضافة إلى الحلقة بعداد هناك نوع آخر من الحلقات نذكر مثلا الحلقة **For ... To ... Do...** و التي تشكل موضوع هذه الفقرة. ما يميز هذه الحلقة عن الحلقة بعداد هو أنها:

1. لا تحتوي على العداد
2. لا تحتوى على اختبار القيمة النهائية للعداد
3. لا تحتوى على مقدار زيادة العداد الممثل في الخطوة

طريقة كتابة الحلقة **For ... To ... Do...** هي التالية:

For Variable_De_Controle **:=** Valeur_Initiale **To** Valeur_Finale **Do** Instruction **;**

حيث أن:

- **For** : و التي تعني من أجل.
- Variable_De_Controle : و التي تعني متغيرة المراقبة و هي عبارة عن متغيرة تكون عادة أو في أغلب المسائل من النوع الصحيح هذه المتغيرة تلعب دور العداد حيث أنها تأخذ قيمة ابتدائية و هي Valeur_Initiale و تأخذ قيمة نهائية هي Valeur_Finale. نشير إلى أنه يجب أن تكون القيمة الابتدائية أقل من القيمة النهائية حتى يتم تنفيذ التعليمة **For ... To ... Do...**
- Valeur_Initiale : تمثل القيمة الابتدائية التي تأخذها متغيرة المراقبة Variable_De_Controle.
- **To** : و التي تعني إلى.
- Valeur_Finale : تمثل القيمة النهائية التي تأخذها متغيرة المراقبة Variable_De_Controle.
- **Do** : و التي تعني أنجز أو نفذ.

- Instruction : تمثل تعليمة واحدة يتم إنجازها أو تنفيذها عدد من المرات [1].
- **;** : فاصلة منقوطة.

يمكن إجراء تعديل شكلي على طريقة الكتابة أعلاه دون أن يؤثر ذلك عليها لتصبح كما يلي:

For Variable_De_Controle **:=** Valeur_Initiale **To** Valeur_Finale **Do**
Instruction **;**

بغرض توضيح كيفية استخدام الحلقة **For … To … Do…** نأخذ المثال التالي:

مثـال 05.09:

نريد نشر على الشاشة 6 مرة ما يلي:

ZAHRA

يتم ذلك باستخدام الحلقة 6 كما يلي:

for i:= 1 to 6 do writeln ('ZAHRA') ;

كما يمكن كتابة ذلك كما يلي:

for i:= 1 to 6 do
writeln ('ZAHRA') ;

من الكتابة أعلاه نستنتج ما يلي:

- المتغيرة المراقبة تتمثل في المتغيرة *i* و هي من النوع الصحيح تأخذ القيمة الابتدائية 1 إلى غاية القيمة النهائية 6
- القيمة الابتدائية تساوي 1
- القيمة النهائية تساوي 6
- التعليمة التي يتم انجازها عدد من المرات يساوي 6 هي التالية:

writeln ('ZAHRA')

الكتابة أعلاه تعني للحاسوب ما يلي:

من أجل القيمة الابتدائية 1 لمتغيرة المراقبة *i* إلى غاية القيمة النهائية 6 أنجز التعليمة التالية:

writeln ('ZAHRA')

يقوم الحاسوب بتنفيذ التعليمة *writeln ('ZAHRA')* عدد من المرات يساوي 6 ثم ينتقل أو يواصل تنفيذذذذ

[1] سوف يتم التطرق لاحقا إلى أنه يمكن تنفيذ أكثر من تعليمة واحدة.

تنفيذ التعليمات التي تليها أي تنفيذ باقي تعليمات البرنامج.

و عليه يصبح البرنامج الكلي هو التالي:

```
program La_BOUCLE_FOR ;
var i : integer ;
begin
for i:= 1 to 6 do
writeln ('ZAHRA');
writeln ;
writeln ('M  E  R  C  I');
readln ;
end.
```

عند تنفيذ الحاسوب للبرنامج أعلاه فإنه يقوم بما يلي:

- تنفيذ 6 مرات التعليمة *writeln ('ZAHRA')* أي نشر 6 مرات *ZAHRA*
- تنفيذ التعليمة و التي تعني تخطي أو قفز سطر واحد
- تنفيذ التعليمة *writeln ('MERCI');* أي نشر *M E R C I*

و عليه فإننا نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
ZAHRA
ZAHRA
ZAHRA
ZAHRA
ZAHRA
ZAHRA

M E R C I
```

مثـال 06.09:

نريد حساب المجموع التالي:

$$som = 5^2 + 6^2 + 7^2 + 8^2 + ... + 27^2$$

يتم حساب المجموع أعلاه باستخدام الحلقة **For … To … Do…** كما يلي:

```
for i:= 5 to 27 do
```

```
som :=som + sqr(i) ;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_FOR ;
var i : integer ;
   som : real ;
begin
som:=0 ;
for i:= 5 to 27 do
som:=som + sqr(i);
writeln ('La Somme :',som);
readln ;
end.
```

مثـال 07.09:

نريد حساب مجموع قيم y من أجل قيم $z = 10,11,12,.....,33$ حيث أن:

$$y = 120z^3 + 50z^2 + 25z + 500$$

يتم حساب المجموع أعلاه باستخدام الحلقة **For … To … Do…** كما يلي:

```
for z:= 10 to 33 do
   somme :=somme + 120*z**3+50*sqr(z)+25*z+500 ;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_FOR ;
var z : integer ;
   somme : real ;
begin
for z:= 10 to 33 do
   somme :=somme + 120*z**3+50*sqr(z)+25*z+500 ;
writeln ('La Somme :',somme);
readln ;
end.
```

ملاحظـة 03.09 :

من خلال جميع الأمثلة أعلاه نلاحظ أن عدد التعليمات التي يتم تنفيذها بعد الحلقة **For …**

To … Do… يساوي تعليمة واحدة. يمكن تنفيذ أكثر من تعليمة واحدة (مجموعة من التعليمات) بعد هذه الحلقة شريطة أن يتم تجميع هذه التعليمات فيما يسمى بكتلة أو مجموعة تعليمات Un Bloc D'instructions أو التعليمة المركبة[1] Une Instruction Composée وفقا لطريقة الكتابة الثانية التالية:

```
For Variable_De_Controle := Valeur_Initiale To Valeur_Finale Do
  Begin
      Instruction_1 ;
      Instruction_2 ;
      Instruction_3 ;
      ……..
      ……..
      ……..
      Instruction_n ;
  End ;
```

بغرض التوضيح نأخذ الأمثلة التالية:

مثـال 08.09:

نريد نشر على الشاشة 12 مرة ما يلي:

Samedi
Dimanche
Mardi

يتم ذلك باستخدام الحلقة **For … To … Do…** كما يلي:

```
for j:= 1 to 12 do
begin
   writeln ('Samedi');
   writeln ('Dimanche');
   writeln ('Mardi');
end ;
```

[1] لمزيد من المعلومات حول التعليمة المركبة أنضر الفقرة 04.07

مثـال 09.09:

نريد حساب ما يلي:

- مربع القيم التالية: 4,5,6,7,8,.....,23
- نشر القيم و المربعات الموافقة لها كما يلي:

Le carré de 4 est : 16

Le carré de 5 est : 25

Le carré de 6 est : 36

……

……

……

Le carré de 23 est : 529

- حساب مجموع القيم : $S = 4+5+6+7+8+.....+23$
- نشر مجموع القيم كما يلي:

La Somme Des Valeurs =

- حساب مجموع مربع القيم : $som = 4^2+5^2+6^2+7^2+8^2+.....+23^2$
- نشر مجموع مربع القيم كما يلي:

La Somme Des Carrés Des Valeurs =

يتم ذالك باستخدام الحلقة **For … To … Do…** كما يلي:

```
for i:= 4 to 23 do
begin
   carre:=sqr(i);
   writeln ('Le carre de  ',i,' est  ',carre);
   s:=s+i;
   som:=som+carre;
end ;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_FOR ;
var i, carre,s,som : integer ;
begin
s:=0;
```

```
som:=0;
for i:= 4 to 23 do
begin
   carre:=sqr(i);
   writeln ('Le carre de  ',i,' est  ',carre);
   s:=s+i;
   som:=som+carre;
end ;
writeln('La Somme Des Valeurs = ',s);
writeln('La Somme Des Carrés Des Valeurs = ',som);
writeln('MERCI');
readln ;
end.
```

ملاحظـة 04.09 :

- يشترط أن تكون القيمة الابتدائية Valeur_Initiale لمتغيرة المراقبة Variable_De_Controle **أقل أو تساوي** القيمة النهائية Valeur_Finale لها.
- إذا كانت القيمة الابتدائية لمتغيرة المراقبة **أكبر** من لقيمة النهائية لها، فإن الحاسوب لا يقوم بتنفيذ الحلقة **For … To … Do…** بل يتجاهلها و يتخطاها لينتقل إلى تنفيذ التعليمات التي تلي الحلقة أي إلى تنفيذ بقية تعليمات البرنامج.

ملاحظـة 05.09 :

- يمكن استخدام طريقة الكتابة الثانية للحلقة **For … To … Do…** حتى في حالة وجود تعليمة واحدة فقط نقوم بتنفيذها مثلما يوضحه المثال أدناه.

مثـال 10.09:

نريد حساب المجموع التالي:

$$some = (-5)^3 + (-4)^3 + (-3)^3 + (-2)^3 + (-1)^3 + \ldots\ldots\ldots + (8)^3$$

يتم ذالك باستخدام الحلقة **For … To … Do…** كما يلي:

```
for i:= -5 to 8 do
```

```
begin
   some:=some+i**3;
end ;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_FOR ;
var i,some : integer ;
begin
some:=0;
for i:= -5 to 8 do
begin
   some:=some+i**3;
end ;
writeln(' some = ',some);
writeln('MERCI');
readln ;
end.
```

من خلال جميع الأمثلة التي تم تناولها أعلاه كان مقدار الزيادة (الإضافة) أو ما يسمى بالخطوة (pas) لمتغيرة المراقبة يساوي القيمة 1 أي أن قيمة متغيرة المراقبة تزداد دائما بالقيمة أو المقدار 1. في لغة الـ Pascal لا يمكن رفع قيمة متغيرة المراقبة بأكثر من المقدار 1 أي مقدار الخطوة يساوي دائما 1 عكس بعض لغات البرمجة الأخرى مثل Basic حيث يمكن التحكم في الخطوة أي تغيير مقدار زيادة متغيرة المراقبة حيث يمكن جعلها تتحرك بخطوة أقل من الواحد أو بخطوة أكبر من الواحد مثل 2 , 3 ، 4 أو أكثر من ذلك. بغرض تجاوز هذا النقص في لغة الـ Pascal المتمثل في ثبات الخطوة عند القيمة 1 نقوم بالاستعانة بمتغيرة إضافية أو أكثر إلى جانب متغيرة المراقبة حيث أن قيمة المتغيرة الإضافية تكون بدلالة متغيرة المراقبة مثلما يوضحه الأمثلة التالية:

مثـال 11.09:

نريد حساب ما يلي:

1. الجداء : $produit = 1+2+3+4+5+6+7+.....+50+51+52$
2. المجموع : $som = 4+8+12+16+20+24+28+.....+200+204+208$

يتم ذالك باستخدام الحلقة **For … To … Do…** كما يلي:

1. بالنسبة للجداء *produit* : نستخدم متغيرة المراقبة *i* حيث أن:
   - القيمة الابتدائية تساوي 1
   - القيمة النهائية تساوي 52
   - الخطوة تساوي 1

و عليه يتم استخدام **For … To … Do…** كما يلي:

```
produit:=1 ;
for i:= 1 to 52 do
begin
   produit:= produit *i;
end ;
```

2. بالنسبة للمجموع *som* : نستخدم متغيرة المراقبة *j* حيث أن:
   - القيمة الابتدائية تساوي 4
   - القيمة النهائية تساوي 208
   - الخطوة تساوي 4 تختلف عن 1

بما أن الخطوة لا تساوي 1 فإنه لا يمكن استخدام الحلقة **For … To … Do…** بمتغيرة المراقبة *j* التي تأخذ القيمة الابتدائية 4 إلى غاية القيمة النهائية تساوي 208 لذالك سوف نحاول إيجاد علاقة بين متغيرة المراقبة *i* ذات الخطوة 1 و المتغيرة *j* . نلاحظ أن بأن قيمة المتغيرة *j* تساوي أربع مرات قيمة متغيرة المراقبة *i* أي أن $j = 4*i$ و بذالك تم حل مشكل الخطوة التي تختلف عن 1. الآن يمكن حساب المجموع *som* باستخدام الحلقة **For … To … Do…** ذات متغيرة المراقبة *i* و العلاقة بين المتغيرتين *i* و *j* أي $j = 4*i$

```
Som:=0 ;
produit:=1 ;
for i:= 1 to 52 do
begin
   produit:= produit *i ;
  j:=4*i ;
 Som:= Som+j ;
end ;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_FOR ;
var i,j,produit,som : integer ;
begin
Som:=0 ;
produit:=1 ;
for i:= 1 to 100 do
begin
   produit:= produit *i ;
  j:=4*i ;
 Som:= Som+j ;
end ;
writeln(' Le Produit  =  ', produit);
writeln(' La Somme  = ',som);
writeln('MERCI');
readln ;
end.
```

مثـال 12.09:

نريد حساب المجاميع التالية:

1. المجموع الأول : $somme = 1+2+3+4+5+.....+99+100$
2. مجموع الثاني : $somme_1 = 1+6+11+16+21+.....+491+496$

يتم ذالك باستخدام الحلقة **For … To … Do…** كما يلي:

1. بالنسبة للمجموع الأول *somme* : نستخدم متغيرة المراقبة *i* حيث أن:
   - القيمة الابتدائية تساوي 1
   - القيمة النهائية تساوي 100
   - الخطوة تساوي 1

و عليه يتم استخدام **For … To … Do…** كما يلي:

```
for i:= 1 to 100 do
begin
   somme:=somme+i;
end ;
```

2. بالنسبة للمجموع الثاني *somme_1* : نستخدم متغيرة المراقبة *j* حيث أن:
   - القيمة الابتدائية تساوي 1
   - القيمة النهائية تساوي 102
   - الخطوة تساوي 5 تختلف عن 1

بما أن الخطوة لا تساوي 1 فإنه لا يمكن استخدام الحلقة **For … To … Do…** بمتغيرة المراقبة *j* التي تأخذ القيمة الابتدائية 1 إلى غاية القيمة النهائية تساوي 102 لذالك سوف نحاول إيجاد علاقة بين متغيرة المراقبة *i* ذات الخطوة 1 و المتغيرة *j* . نلاحظ أن بأن قيمة المتغيرة *j* تفوق قيمة متغيرة المراقبة *i* حيث أن $j=1+5*(i-1)$ و بذالك تم حل مشكل الخطوة التي تختلف عن 1. الآن يمكن حساب المجموع *somme_1* باستخدام الحلقة **For … To … Do…** ذات متغيرة المراقبة *i* و العلاقة بين المتغيرتين *i* و *j* أي $j=1+5*(i-1)$

```
for i:= 1 to 100 do
begin
   somme:=somme+i;
   j :=1+5*(i-1) ;
  somme_1 :=somme_1+j ;
end ;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_FOR ;
var i,j,somme,somme_1 : integer ;
begin
somme:=0;
somme_1:=0;
for i:= 1 to 100 do
begin
   somme:=somme+i;
   j :=1+5*(i-1) ;
   somme_1 :=somme_1+j ;
end ;
writeln(' somme = ',somme);
```

```
writeln(' somme_1 = ',somme_1);
writeln('MERCI');
readln ;
end.
```

مثـال 13.09:

نريد حساب الجداء التالي:

$$produit = 3^0 * 4^1 * 5^2 * 6^3 * 7^4 * 8^5 * ..... * 20^{17}$$

يتم ذالك باستخدام الحلقة **For … To … Do…** كما يلي:

يتم استخدام متغيرتين مراقبة i و j إحداهما لها علاقة بالثانية حيث أن:

- i تأخذ القيمة الابتدائية 0 إلى غاية القيمة النهائية 17 و $j = i + 3$
- i تأخذ القيمة الابتدائية 3 إلى غاية القيمة النهائية 20 و $j = i - 3$

و عليه يصبح البرنامج الذي يسمح بحساب الجداء أعلاه هو التالي:

```
program La_BOUCLE_FOR ;
var i,j,produit : integer ;
begin
produit:=1;
for i:= 0 to 17 do
begin
    j :=i+3 ;
   produit :=produit*(j)**i ;
end ;
writeln(' Le Produit = ',produit);
writeln('MERCI');
readln ;
end.
```

ملاحظـة 06.09 :

في بعض الحالات و المسائل متغيرة المراقبة تأخذ قيمة ابتدائية كبيرة و قيمة نهائية صغيرة مثلما توضحه الأمثلة أدناه . في هذه الحالة لا يتم استخدام الحلقة **For … To … Do…** و إنما يتم استخدام الحلقة **For … Downto … Do…**

طريقة كتابة الحلقة **For … Downto … Do…** هي التالية:

```
For Variable_De_Controle := Valeur_Initiale Downto Valeur_Finale Do
    Instruction ;
```

أما في حالة وجود العديد من التعليمات المراد تنفيذها، فإن طريقة الكتابة هي التالية:

```
For Variable_De_Controle := Valeur_Initiale Downto Valeur_Finale Do
  Begin
      Instruction_1 ;
      Instruction_2 ;
      Instruction_3 ;
      ……..
      ……..
      ……..
      Instruction_n ;
  End ;
```

بغرض التوضيح نأخذ الأمثلة التالية:

مثـال 14.09:

نريد حساب : $somme = 100 + 99 + 98 + 97 + 96 + ... + 2 + 1 + 0$

بغرض حساب المجموع أعلاه نستخدم متغيرة المراقبة j حيث أن:

- القيمة الابتدائية تساوي 100
- القيمة النهائية تساوي 0
- مقدار نقصان متغيرة المراقبة j يساوي 1 أي أن الخطوة تساوي -1

يتم استخدام **For … Downto … Do…** كما يلي:

```
for j:= 100 downto 0 do
begin
   somme:=somme+j;
end ;
```

مثـال 15.09:

نريد حساب : $factoriel = 20*19*18*17*.....*3*2*1$

يتم استخدام **For … Downto … Do…** كما يلي:

```
Factoriel:=1;
for j:= 20 downto 1 do
begin
   factoriel:=factoriel*j;
end ;
```

ملاحظـة 07.09 :

- يشترط أن تكون القيمة الابتدائية Valeur_Initiale لمتغيرة المراقبة Variable_De_Controle **أكبر أو تساوي** القيمة النهائية Valeur_Finale لها.
- إذا كانت القيمة الابتدائية لمتغيرة المراقبة **أقل** من لقيمة النهائية لها، فإن الحاسوب لا يقوم بتنفيذ الحلقة **For … Dwnto … Do…** بل يتجاهلها و يتخطاها لينتقل إلى تنفيذ التعليمات التي تلي الحلقة أي إلى تنفيذ بقية تعليمات البرنامج.

06.09 – الحلقـة … *until* … *Repeat* La Boucle:

إضافة إلى الحلقة *For ... to ... Do...* التي تسمح بتنفيذ تعليمة أو مجموعة من التعليمات عدد **معلوم** من المرات ليتم بعد ذلك الخروج من الحلقة، هناك الحلقة *Repeat ... until ...* التي تشكل موضوع هذه الفقرة.

الحلقة *Repeat ... until ...* كذالك تسمح بتنفيذ تعليمة أو مجموعة من التعليمات عدد **مجهول** من المرات حتى يتحقق الشرط أي حتى يصبح الشرط محقق.

طريقة كتابة الحلقة *Repeat ... until ...* هي التالية:

```
Repeat
        Instruction_1 ;
        Instruction_2 ;
        Instruction_3 ;
        ……..
```

……..

……..

Instruction_n **;**

Until Condition **;**

حيث أن:

- repeat : و التي تعني إعادة .
- Instruction_n …… Instruction_1 : تمثل التعليمات التي نريد تنفيذها في حالة عدم تحقق الشرط condition .
- until : حتى يتحقق الشرط condition.
- condition : تمثل شرط .

بغرض توضيح كيفية استخدام الحلقة *Repeat ... until ...* نأخذ الأمثلة التالية:

مثـال 16.09:

نريد باستخدام الحلقة *Repeat ... until ...* إدخال عن طريق لوحة المفاتيح قيمة موجبة للمتغيرة من النوع الصحيح x أي أن $x > 0$ بحيث يطلب إعادة إدخال قيمة x في حالة أن $x \leq 0$

يتم ذالك كما يلي:

```
repeat
   write ('Donnez Une Valeur Positive Pour X : ');
   readln(x);
until x > 0  ;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLES_repeat_until;
var x :integer ;
begin
repeat
   write ('Donnez Une Valeur Positive Pour X : ');
   readln(x);
until x > 0  ;
writeln ('La Valeur Introduite est ',x);
readln ;
```

end.

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

النشر على الشاشة

```
Donnez Une Valeur Positive Pour X :  0
Donnez Une Valeur Positive Pour X : -14
Donnez Une Valeur Positive Pour X : -1
Donnez Une Valeur Positive Pour X : 1
La Valeur Introduite est  1
_
```

في هذا المثال يقوم الحاسوب بإعادة تنفيذ التعليمتين المحصورتين بين *Repeat* و *until* إلى غاية أو حتى تحقق الشرط $x > 0$ أي حتى يتم إدخال قيمة موجبة للمتغيرة الصحيحة x.

ملاحظـة 08.09 :

- مع الحلقة ... *Repeat ... until* لا نعلم مسبقا عدد مرات تنفيذ الحلقة أو التعليمات التي تحتويها الحلقة على عكس الحلقة ...*For ... to ... Do* أين نعلم مسبقا عدد مرات تنفيذ تعليمات هذه الحلقة.
- يقوم الحاسوب باختبار تحقق الشرط أم **بعد** الانتهاء من تنفيذ الحلقة *Repeat ... until* ...
- لا نعلم عدد مرات تنفيذ تعليمات الحلقة ... *Repeat ... until* و لكن الأكيد أن تعليمات تنفذ على الأقل مرة واحدة حتى و لو كان الشرط Condition غير محقق.
- يتم الخروج من الحلقة ... *Repeat ... until* بعد تنفيذ جميع تعليماتها في حالة واحدة فقط و هي تحقق الشرط Condition، في حالة عدم تحقق الشرط يعيد الحاسوب تنفيذ التعليمات التي تحتويها الحلقة.

مثـال 17.09:

نريد إدخال مجموعة من الأعداد (عددها مجهول) حتى يكون مجموعها يساوي أو يفوق 1500 يتم ذالك باستخدام الحلقة ... *Repeat ... until* كما يلي:

```
repeat
    write ('Donnez Une Valeur : ');
    readln(valeur);
    somme:=somme+valeur;
```

```
until somme >= 1500  ;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_repeat_until;
var valeur,somme :integer ;
begin
somme:=0;
repeat
    write ('Donnez Une Valeur : ');
    readln(valeur);
    somme:=somme+valeur;
until somme >= 1500  ;
writeln ('La Somme = ',somme);
readln ;
end.
```

07.09 – الحلقة ... *While ... do* ... La Boucle:

إضافة إلى الحلقتين ... *For ... to ... Do* و ... *Repeat ... until* ، هناك حلقة ثالثة هي ... *While ... do* و التي تشكل موضوع هذه الفقرة.

مثلها مثل الحلقتين السابقتين، فإن الحلقة ... *While ... do* كذالك تسمح بتنفيذ تعليمة أو مجموعة من التعليمات عدد **مجهول** من المرات مادام الشرط محقق.

طريقة الكتابة الأولى للحلقة ... *While ... do* هي التالية:

```
While Condition Do  Instruction ;
```

هذا في حالة وجود تعليمة واحدة يتم تنفيذها أما في حالة وجود أكثر من تعليمة يتم تنفيذها، فإن لها طريقة كتابة ثانية هي التالية:

```
While Condition Do
  Begin
      Instruction_1 ;
      Instruction_2 ;
```

```
        Instruction_3 ;
        ……..
        ……..
        ……..
        Instruction_n ;
End ;
```

حيث أن:

- While : و التي تعني مادام .
- condition : تمثل شرط .
- Instruction_n …… Instruction_1 : تمثل التعليمات التي نريد تنفيذها في حالة تحقق الشرط condition .
- Do : أنجز أو نفذ.

بغرض توضيح كيفية استخدام الحلقة ... *While ... do* نأخذ الأمثلة التالية:

مثـال 18.09:

نريد باستخدام الحلقة ... *While ... do* إدخال عن طريق لوحة المفاتيح قيمة موجبة للمتغيرة من النوع الصحيح x أي أن $x > 0$ بحيث يطلب إعادة إدخال قيمة x في حالة أن $x \leq 0$

يتم ذالك كما يلي:

```
x:=0;
while x<= 0 do
begin
  write (' Donnez Une Valeur Positive Pour  X: ');
  readln(x);
end;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_While_Do;
var x :integer ;
begin
x:=0;
while x<= 0 do
```

```
begin
  write (' Donnez Une Valeur Positive Pour  X: ');
  readln(x);
end;
writeln('La Valeur Introduite est ',x);
readln ;
end.
```

في هذا المثال يقوم الحاسوب بإعادة تنفيذ التعليمة المركبة المكونة من التعليمتين:

```
begin
  write (' Donnez Une Valeur Positive Pour  X: ');
  readln(x);
end;
```

مادام الشرط ***x<=0*** محقق أي مادام لم يتم إدخال قيمة موجبة أو مادامت قيمة *x* معدومة أو سالبة

ملاحظـة 09.09 :

- مع الحلقة *While ... do ...* كذالك لا نعلم مسبقا عدد مرات تنفيذ الحلقة مثلما هو الحال مع الحلقة *Repeat ... until ...* و على عكس الحلقة *For ... to ... Do...* أين نعلم مسبقا عدد مرات تنفيذ تعليمات هذه الحلقة.
- يقوم الحاسوب باختبار تحقق الشرط أم لا **قبل** البدء في تنفيذ الحلقة بالنسبة للحلقة *While ... do ...* و **بعد** الانتهاء من تنفيذ الحلقة بالنسبة للحلقة *Repeat ... until ...*
- يمكن للحاسوب أن لا ينفذ أبدا الحلقة *While ... do ...* و هذا فقط في حالة عدم تحقق الشرط Condition لأن مع هذه الحلقة يقوم الحاسوب باختبار تحقق الشرط أم لا **قبل** البدء في تنفيذها على عكس الحلقة *Repeat ... until ...* التي تنفذ على الأقل مرة واحدة حتى و لو كان الشرط Condition غير محقق لأن مع هذه الحلقة يقوم الحاسوب باختبار تحقق الشرط أم لا **بعد** الانتهاء من تنفيذها.
- الشرط الذي بعد *While* يمثل أو عبارة عن شرط **مواصلة** تنفيذ الحلقة *While ... do ...* أما الشرط الذي بعد فيمثل أو عبارة عن شرط **توقف** عن تنفيذ الحلقة أي شرط الخروج من الحلقة.

مثـال 19.09:

نريد إدخال مجموعة من الأعداد (عددها مجهول) حتى يكون مجموعها يساوي أو يفوق 1500

يتم ذالك باستخدام الحلقة ... *While ... do* كما يلي:

```
some:=0;
while some <= 1500 do
begin
  write (' Donnez Une Valeur : ');
  readln(valeur);
  some:=some+valeur;
end;
```

و عليه يصبح البرنامج هو التالي:

```
program La_BOUCLE_While_Do;
var valeur ,some :integer ;
begin
some:=0;
while some <= 1500 do
begin
  write (' Donnez Une Valeur : ');
  readln(valeur);
  some:=some+valeur;
end;
writeln('La SOMME = ',some);
readln ;
end.
```

08.09 – الحلقات المتداخلة Les Boucles Imbriquées:

يمكن لبرنامج أن يتضمن أكثر من حلقة أي يحتوي على العديد من الحلقات مثلما يوضحه المثال التالي:

مثال 20.09:

حساب العمليات التالية:

1. $som_1 = -5 + -4 + -3 + -2 + -1 + 0 + 1 + 2 + 3 + ... + 173 + 171 + 175$
2. $som_2 = 10 + 11 + 12 + 13 + 14 + 15 + 16 + ... + 198 + 199 + 200$

3. $som_3 = -200 - 199 - 198 - 197 - 196 - ..... - 102 - 101 - 100$

البرنامج الذي يسمح بذلك هو التالي:

```
program TROIS_BOUCLES;
var i,j,k,som_1,som_2,som_3 : integer ;
begin
{ Programme 1ere Opération : 1ERE BOUCLE}
som_1:=0 ;
for i:=-5 to 175 do
som_1:=som_1+i ;
writeln ('some_1 = ',som_1);
{ Programme 2eme Opération : 2 EME BOUCLE}
som_2:=0 ;
for j:=10 to 200 do
som_2:=som_2+j ;
writeln ('some_2 = ',som_2);
{ Programme 3eme Opération : 3 EME BOUCLE}
som_3:=0 ;
for k:=-200 to -100 do
som_3:=som_3+k ;
writeln ('some_3 = ',som_3);
readln ;
end.
```

كما يمكن لحلقة أن تتضمن أو تحتوي على حلقة أو أكثر، في هذه الحالة نقول أن الحلقات متداخلة مثلما يوضحه المثال التالي:

مثـال 21.09:

نريد حساب نتيجة العملية التالية:

- $som_fact = 5!+6!+7!+8!+9!+10!+..... + 20!+21!+22!$

العملية تحتوي على عمليتين هما:

1. عملية حساب العاملي للأعداد من 5 إلى 22

البرنامج الذي يسمح بحساب العاملي لأي عدد x هو التالي:

$$x!= x*(x-1)*(x-2)*(x-3)*(x-4)*.....*4*3*2*1$$

$$x!=1*2*3*4*.....*(x-4)*(x-3)*(x-2)*(x-1)*x$$

```
fact:=1 ;
for i:=1 to x do
fact:=fact * i ;
```

2. عملية حساب المجموع لنتيجة العاملي

و عليه يصبح البرنامج هو التالي:

```
program DEUX_BOUCLES_IMBRIQEES ;
var i,j : integer ;
   som_fact,fact : real ;
begin
som_fact:=0 ;
for j:=5 to 22 do
  begin
    fact:=1 ;
    for i:=1 to j do
        fact:=fact * i ;
    som_fact:=som_fact + fact ;
  end;
writeln ('La Somme = ',som_fact) ;
readln ;
end.
```

مثـال 22.09:

نريد حساب و نشر قيم المتغيرة z حيث أن:

$$z = 10x^2 + 20y^2 - 5xy + 150$$

حيث أن:

- $x = 1,2,3,...,20$
- $y = 10,11,1213,14,15$

البرنامج الذي يسمح بذالك هو التالي:

```
program DEUX_BOUCLES_IMBRIQEES ;
var x,y,z : integer ;
begin
for x:=1 to 20 do
  begin
    for y:=10 to 15 do
        begin
          z:=10*sqr(x)+20*sqr(y)-5*x*y+150 ;
          writeln ( z)
        end;
  end;
readln ;
end.
```

تمارين الفصل

✍ التمرين 01.09:

1. ماذا نعني بالمعالجة التكرارية Le Traitement Répititif
2. ماذا نعني بالحلقة La Boucle
3. ماذا نعني بالخطوة Le Pas
4. أذكر جميع أنواع الحلقات

✍ التمرين 02.09:

نريد حساب و نشر قيمة المتغيرة z حيث أن:

$$z = 10X^2 + 5X - 15$$

حيث أن $X = 1,2,3,...,100$

– **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيم z من أجل كل قيمة من قيم المتغيرة X و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 03.09:

ليكن *somme* حيث أن:

$$somme = 1+2+3+4+5+.....+120$$

– **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة *somme* و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 04.09:

ليكن s حيث أن:

$$s = 1^2 + 2^2 + 3^2 + 4^2 + ... + 200^2$$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة s و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 05.09:

ليكن som حيث أن:

$$som = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} ..... + \frac{1}{150}$$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة som و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 06.09:

ليكن $som1$ حيث أن:

$$som1 = \frac{1}{150^2} + \frac{1}{149^2} + \frac{1}{148^2} + ... + \frac{1}{5^2} + \frac{1}{4^2} + \frac{1}{3^2} + \frac{1}{2^2} + 1$$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة $som1$ و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…

- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 07.09:

ليكن *produit* حيث أن:

$$produit = 2 \times 3 \times 4 \times 5 \times ..... \times 48 \times 49 \times 50$$

- **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة *produit* و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 08.09:

ليكن *prod* حيث أن:

$$prod = 150 \times 149 \times 148 \times 147 \times ..... \times 4 \times 3 \times 2$$

- **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة *prod* و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 09.09:

ليكن $prod_1$ حيث أن:

$$prod_1 = 1 \times 3 \times 5 \times ..... \times 195 \times 197 \times 199$$

- **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة $prod_1$ و هذا باستخدام ما يلي:

- حلقة بعداد

- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 10.09:

نود الحصول على الشاشة على ما يلي:

```
*
**
***
****
*****
******
*******
********
*********
**********
```

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بالحصول على الشكل أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 11.09:

نود الحصول على الشاشة على ما يلي:

```
**********
*********
********
*******
******
*****
****
***
**
*
```

– **المطلوب:**

1. كتابة البرنامج الذي يسمح بالحصول على الشكل أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 12.09:

1. أكتب البرنامج الذي يقوم بحساب مجموع الأعداد الأولى حتى يساوي أو يفوق 200000

$$s = 1^2 + 2^2 + 3^2 + 4^2 + ... + ... \geq 200000$$

و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 13.09:

ليكن *somme* حيث أن:

$$somme = 1 + \frac{1}{3^3} + \frac{1}{5^3} + \frac{1}{7^3} + ... + \frac{1}{21^3}$$

– **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة *somme* و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 14.09:

1. أكتب البرنامج الذي يقوم بحساب أس (قوة) عدد و ليكن n حيث أن:

$$n^m = n \times n \times n \times n \times n \times ..... \times n$$

و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 15.09:

$n!$ يساوي ما يلي:

$$n!=n\times(n-1)\times(n-2)\times(n-3)\times(n-4)\times.....\times1$$

$$n!=1\times2\times3\times4\times5\times.....\times(n-4)\times(n-3)\times(n-2)\times(n-1)\times1$$

– **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة $n!$ و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 16.09:

$(n-r)!$ يساوي ما يلي:

$$(n-r)!=(n-r-1)\times(n-r-2)\times(n-r-3)\times(n-r-4)\times.....\times1$$

– **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة $(n-r)!$ و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 17.09:

1. أكتب البرنامج الذي يسمح بحساب ما يلي:

$$\frac{n!}{(n-r)!}$$

و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 18.09:

1. أكتب البرنامج الذي يسمح بحساب ما يلي:

$$\frac{n!}{r!(n-r)!}$$

و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 19.09:

e^x يساوي ما يلي:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \ldots\ldots\ldots + \frac{x^n}{n!}$$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة e^x و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 20.09:

أكتب البرنامج الذي يسمح بـ:

1. إدخال 5 علامات $not_1, note_2, \ldots, note_5$ كما يلي:

Introduire Note Numéro 1 :

Introduire Note Numéro 2 :

Introduire Note Numéro 3 :

Introduire Note Numéro 4 :

Introduire Note Numéro 5 :

2. حساب المجموع العلامات *som* حيث أن:

$$som = not_1 + note_2 + ... + note_5$$

3. حساب المعدل

$$moy = \frac{not_1 + note_2 + ... + note_5}{5}$$

4. نشر المعدل

و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 21.09:

1. أكتب البرنامج الذي يسمح بحساب مجموع 50 عدد زوجي الأولى

$$2 + 4 + 6 + 8 + 10 + ...$$

و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 22.09:

1. أكتب البرنامج الذي يسمح بحساب مجموع 50 عدد فردي الأولى

$$1 + 3 + 5 + 7 + 9 + ...$$

و هذا باستخدام ما يلي:

- حلقة بعداد

- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 23.09:

1. أكتب البرنامج الذي يسمح بحساب $\frac{\pi^2}{6}$ حيث أن:

$$\frac{\pi^2}{6}=\frac{1}{1^2}+\frac{1}{2^2}+\frac{1}{3^2}+\frac{1}{4^2}+.....+\frac{1}{100^2}$$

و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 24.09:

أكتب البرنامج الذي يسمح بـ:

1. إدخال 10 قيم للمتغير X و 10 قيم للمتغير N كما يلي:

Introduire X1:
Introduire N1:
Introduire X2:
Introduire N2:
Introduire X3:
Introduire N3:
……..
……..
Introduire X10:
Introduire N10:

2. حساب و نشر مجموع قيم المتغير X
3. حساب و نشر مجموع قيم المتغير N
4. حساب و نشر مجموع جداء قيم المتغير X بالمتغير N

✍ التمـريـن 25.09:

نريد حساب و نشر قيمة المتغيرة y حيث أن:

$$y = 5X^3 + 2X^2 - 3X - 4$$

حيث أن X يتغير من 10 إلى -6 بخطوة تساوي -2.5

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيم y من أجل كل قيمة من قيم المتغيرة X و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 26.09:

ليكن y حيث أن:

$$y = e^{-2x}$$

حيث أن X يتغير من 3 إلى -1 بخطوة تساوي -0.5

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيم y من أجل كل قيمة من قيم المتغيرة X و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 27.09:

ليكن y حيث أن:

$$y = 5X + 10$$

حيث أن X يتغير من 10 إلى -10 بخطوة تساوي -1

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيم y حيث أن $y \leq 20$ من أجل كل قيمة من قيم المتغيرة X و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 28.09:

أكتب البرنامج الذي يسمح بـ:

1. إدخال 5 علامات لـ 10 طلبة
2. حساب مجموع النقاط لكل طالب
3. حساب المعدل لكل طالب
4. نشر النتائج كما يلي:

NOM	PRENOM	MOYENNE
YA	ZAHRA	18.00
……..	…….	…….
……..	…….	…….
……..	…….	…….
MENAD	SIHEM	17.50

✍ التمـريـن 29.09:

ليكن البرنامج التالي:

```
program BOUCLES ;
var i,j : integer ;
begin
    j:=5 ;
    for i := 1 to  3 do
    j:=j+1 ;
    writeln ('La Valeur De j : ',j);
```

```
end.
```

- **المطلوب:**

1. عند تنفيذ البرنامج أعلاه، ما هي القيمة النهائية للمتغيرة *j*

✍ التمرين 30.09:

ليكن البرنامج التالي:

```
program BOUCLES ;
var i,j : integer ;
begin
    j:=1 ;
    for i := 10 downto  3 do
    j:=j+1 ;
    writeln ('La Valeur De j : ',j);
    i:=i+1 ;
    writeln ('La Valeur De i : ',i);
end.
```

- **المطلوب:**

1. عند تنفيذ البرنامج أعلاه، ما هي القيمة النهائية للمتغيرة *j*

✍ التمرين 31.09:

ليكن البرنامج التالي:

```
program BOUCLES ;
var i,j : integer ;
begin
    j:=5 ;
    for i := 1 to  10 do
    j:=j+1 ;
    writeln ('La Valeur De j : ',j);
    writeln ('La Valeur De i : ',i);
end.
```

- **المطلوب:**

1. عند تنفيذ البرنامج أعلاه، ما هي القيمة النهائية للمتغيرة j

✍ التمرين 32.09:

ليكن البرنامج التالي:

```
program BOUCLES ;
var i,j,l : integer ;
begin
    j:=45 ;
    for i := 1 to  10 do
    for l:= 2 to 8 do
    j:=j+i+l;
    writeln ('La Valeur De j : ',j);
    writeln ('La Valeur De i : ',i);
    writeln ('La Valeur De l : ',l);
end.
```

- **المطلوب:**

1. عند تنفيذ البرنامج أعلاه، ما هي القيمة النهائية للمتغيرة j

✍ التمرين 33.09:

ليكن z حيث أن :

$$z = \frac{1}{x} + \frac{1}{x+y} + \frac{1}{x+3y} + \frac{1}{x+5y} + ... + \frac{1}{x+99y}$$

- **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة z حيث أن من أجل كل قيم x و y هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

التمـريـن 34.09:

أكتب البرنامج الذي يسمح بـ:

1. إدخال 10 قيم للمتغير X و 10 قيم للمتغير Y كما يلي:

Introduire X1:
Introduire Y1:
Introduire X2:
Introduire Y2:
Introduire X3:
Introduire Y3:
……..
……..
Introduire X10:
Introduire Y10:

2. حساب المتوسط الحسابي للمتغير $\overline{X}$

$$\overline{X} = \frac{\sum_{i=1}^{10} X_i}{10}$$

3. حساب المتوسط الحسابي للمتغير $\overline{Y}$

$$\overline{X} = \frac{\sum_{i=1}^{10} X_i}{10}$$

4. نشر المتوسط الحسابي $\overline{X}$
5. نشر المتوسط الحسابي $\overline{Y}$

التمـريـن 35.09:

الجدول أدناه يقدم علامات و معاملات خمس مقاييس

N° Du Module	Module	Note	Coefficient
1	STAT_1	17.25	2
2	STAT_2	11.00	3
3	MATHS_1	12.75	3
4	MATHS_2	15.00	3
5	INFORMATIQUE	13.25	2

– **المطلـوب:**

كتابة البرنامج الذي يسمح بحساب:

1. المتوسط الحسابي المرجح MOY_A حيث أن:

$$MOY_A = \frac{\sum Note \times Coefficient}{\sum Coefficient}$$

2. المتوسط الهندسي المرجح MOY_G حيث أن:

$$MOY_G = \sqrt[\sum Coefficient]{\prod Note^{Coefficient}}$$

3. المتوسط التوافقي المرجح MOY_H حيث أن:

$$MOY_H = \frac{\sum Coefficient}{\sum \frac{Coefficient}{Note}}$$

و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 36.09:

ليكن البرنامج التالي:

```
program BOUCLES ;
var i,j,l : integer ;
begin
    for i := 1 to 5 do
    begin
        writeln (i) ;
        for j := 1 to 10 do
        write (j) ;
        writeln ;
```

```
end;
end.
```

- **المطلوب:**

1. عند تنفيذ البرنامج أعلاه، ما ذا نحصل على الشاشة

✍ التمرين 37.09:

لتكن المجاميع التالية:

3. مجموع الأول : $somme = 1+2+3+4+5+.....+99+100$
4. مجموع الثاني : $somme_1 = 1+6+11+16+21+.....+491+496$

- **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر المجاميع أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمرين 38.09:

لتكن العمليات التالية:

1. العملية الأولى : $factoriel = 20*19*18*17*.....*3*2*1$
2. العملية الثانية : $somme = 100+99+98+97+96+.....+2+1+0$
3. العملية الثلثة : $xxx = 50-49-48-47-46-.....-2-1-0$

- **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة العمليات أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 39.09:

لتكن العمليتين التاليتين:

1. العملية الأولى: $produit = 3^0 * 4^1 * 5^2 * 6^3 * 7^4 * 8^5 * ..... * 20^{17}$

2. العملية الثانية: $produit = 20^{17} * ..... * 7^4 * 6^3 * 5^2 * 4^1 * 3^0$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة العمليتين أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 40.09:

نريد حساب المجاميع التالية:

1. المجموع الأول : $somme = 1+2+3+4+5+.....+99+100$
2. المجموع الثاني : $somme_1 = 1+6+11+16+21+.....+491+496$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة العمليات أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 41.09:

نريد حساب ما يلي:

3. الجداء : $produit = 1+2+3+4+5+6+7+.....+50+51+52$
4. المجموع : $som = 4+8+12+16+20+24+28+.....+200+204+208$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة العمليات أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 42.09:

نريد حساب المجموع التالي:

$$some = (-5)^3 + (-4)^3 + (-3)^3 + (-2)^3 + (-1)^3 + ... + (8)^3$$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة العمليات أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 43.09:

نريد حساب ما يلي:

- مربع القيم التالية: 4,5,6,7,8,.....,23
- نشر القيم و المربعات الموافقة لها كما يلي:

Le carré de 4 est : 16

Le carré de 5 est : 25

Le carré de 6 est : 36

……

……

……

Le carré de 23 est : 529

- حساب مجموع القيم : $S = 4+5+6+7+8+.....+23$

- نشر مجموع القيم كما يلي:

La Somme Des Valeurs =

- حساب مجموع مربع القيم : $som = 4^2 + 5^2 + 6^2 + 7^2 + 8^2 + ... + 23^2$
- نشر مجموع مربع القيم كما يلي:

La Somme Des Carrés Des Valeurs =

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة العمليات أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 44.09:

نريد نشر على الشاشة 12 مرة ما يلي:

Samedi
Dimanche
Mardi

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة ما هو أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 45.09:

نريد حساب نتائج العمليات التالية:

3. $S = 1 + 1.5 + 2 + 2.5 + 3 + 3.5 + 4 + 4.5 + ... + 100 + 100.5$

4. $y = 200x + 15$ حيث أن $x = 5,10,15,20,25,30,35,.....,95,100,105$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة ما هو أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 46.09:

لتكن العملية الحسابية التالية:

$$som_fact = 5!+6!+7!+8!+9!+10!+.....+20!+21!+22!$$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر نتيجة العملية أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 47.09:

نريد حساب و نشر قيمة المتغيرة z حيث أن:

$$z = 10x^2 + 20y^2 - 5xy + 150$$

حيث أن:

- $x = 1,2,3,...,10$
- $y = 10,11,1213,14,15$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة المتغيرة z و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 48.09:

حساب العمليات التالية:

4. $som_1 = -5 + -4 + -3 + -2 + -1 + 0 + 1 + 2 + 3 + ... + 173 + 171 + 175$
5. $som_2 = 10 + 11 + 12 + 13 + 14 + 15 + 16 + ... + 198 + 199 + 200$
6. $som_3 = -200 - 199 - 198 - 197 - 196 - ..... - 102 - 101 - 100$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب العمليات أعلاه و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 49.09:

نريد حساب و نشر قيمة المتغيرة z حيث أن:

$$z = 2x^2 + 5y^2 - 15l^2 - 5xyl + 15xl + 60$$

حيث أن:

- $x = 1,2,3,...,10$
- $y = 10,11,1213,14,15$
- $l = 0,1,23,4,5$

– **المطلـوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة المتغيرة z و هذا باستخدام ما يلي:

- حلقة بعداد
- الحلقة while… do…
- الحلقة repeat… until…
- الحلقة for… to…do…

✍ التمـريـن 50.09:

نريد إدخال عن طريق لوحة المفاتيح قيمة موجبة للمتغيرة من النوع الصحيح x أي أن $x > 0$ بحيث يطلب إعادة إدخال قيمة x في حالة أن $x \leq 0$

– **المطلوب:**

1. كتابة البرنامج الذي يسمح بحساب و نشر قيمة المتغيرة z و هذا باستخدام ما يلي:

- الحلقة while… do…
- الحلقة repeat… until…

الفصـــل 📖 العـــاشـــر

الجداول وحيدة البعد (الأشعة)

Les Tableaux Unidimensionnels (Les Vecteurs)

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- ماذا نعني بالجداول وحيدة البعد (الأشعة).
- بعد الجداول وحيدة البعد (الأشعة).
- كيفية التصريح بالجداول وحيدة البعد (الأشعة).
- مختلف الطرق المستخدمة في إدخال معطيات الجداول وحيدة البعد (الأشعة).
- كيفية إدخال معطيات الجداول وحيدة البعد (الأشعة).
- مختلق العمليات الممكن إنجازها على الجداول وحيدة البعد (الأشعة).

01.10 – تمهيـد:

تم التطرق من خلال الفصول الأولى إلى المتغيرات بجميع أنواعها (حقيقية، صحيحة، حرف، سلسلة أحرف)، ما يميز هذه المتغيرات أنها تأخذ أو تساوي قيمة واحدة، هناك نوع آخر يأخذ أكثر من قيمة واحدة أي العديد من القيم **من نفس النوع**، هذا النوع الآخر يسمى شعاع Vecteur أو جدول وحيد البعد Tableau Unidimensionnel و الذي يشكل موضوع هذا الفصل.

02.10 – التعريف بالجداول وحيدة (أحادية) البعد (الأشعة):

الشعاع عبارة عن جدول يتضمن مجموعة من القيم أو العناصر، عدد هذه القيم يسمى بعد Dimension الشعاع حيث نميز نوعين من الأشعة هما:

1. شعاع عمود: الشعاع Z ذو البعد 5 يأخذ الشكل التالي:

$$Z=\begin{pmatrix}3\\2\\1\\6\\7\end{pmatrix}$$

2. شعاع سطر: الشعاع W ذو البعد 8 يأخذ الشكل التالي:

$$W=(23\quad 4\quad 78\quad 9\quad 8\quad 10\quad 11\quad 12)$$

يتم تمثيل أي شعاع و ليكن V ذو البعد 7 سواء كان سطر أو عمود على مستوى الذاكرة كما يلي:

V	15	4	89	4	315	123	20
	$V[1]$	$V[2]$	$V[3]$	$V[4]$	$V[5]$	$V[6]$	$V[7]$

انطلاقا من التمثيل أعلاه للشعاع V يمكن اعتبار أن الجدول وحيد البعد (الشعاع) عبارة عن مجموعة من الخانات المتجاورة في الذاكرة، هذه الخانات تحمل نفس الاسم و الذي هو اسم الشعاع الذي هو V و ذات مؤشرات مختلفة حيث أن المؤشر و ليكن i يتغير من 1 إلى 7 أي $i=1,2,3,.....,7$

- $V[1]$: تمثل الخانة ذات الرتبة رقم $i=1$ في الشعاع V و هي تحتوي على القيمة 15 أي $V[1]=15$
- $V[2]$: تمثل الخانة ذات الرتبة رقم $i=2$ في الشعاع V و هي تحتوي على القيمة 4 أي $V[2]=4$
- .....
- .....
- .....

- $V[7]$: تمثل الخانة ذات الرتبة رقم $i=7$ في الشعاع V و هي تحتوي على القيمة 20 أي $V[7]=20$

03.10 – بعد الجداول وحيدة (أحادية) البعد (الأشعة):

الذي يحدد بعد الشعاع هو عدد العناصر التي يتضمنها أو يحتويها الشعاع، و بالتالي نسمي بعد (Dimension) الشعاع عدد العناصر أو القيم التي يحتويها هذا الشعاع.

مثـال 01.10:

- الشعاع A يحتوي على 9 عناصر أو قيم و بالتالي فإن بعده يساوي 9 و يمثل كما يلي:

A	24	0	345	98	67	1	67	2	1058
	$A[1]$	$A[2]$	$A[3]$	$A[4]$	$A[5]$	$A[6]$	$A[7]$	$A[8]$	$A[9]$

- الشعاع B يحتوي على 50 عنصر أو قيمة و بالتالي فإن بعده يساوي 50 و يمثل كما يلي:

B	5	159	……..	……..	……..	……..	……	96	132
	$B[1]$	$B[2]$	……..	……..	……..	……..	……	$B[49]$	$B[50]$

04.10 – التصريح بالجداول وحيدة (أحادية) البعد (الأشعة):

مثلما يتم التصريح بأي متغيرة قبل استخدامها في البرنامج، فإنه كذالك يتم التصريح بالجداول وحيدة البعد حتى نتمكن من استخدامها في البرنامج. عملية التصريح هذه تتم بواسطة Le mot réservé **ARRAY** وفقا لطريقة الكتابة التالية:

Var** Nom_Du_Tableau **: array [1.. n] of** Type_ Du_Tableau **;

حيث أن:

- ***Var*** : يمثل Le Mot Reservé الذي يسمح بالتصريح باسم الشعاع (الجدول وحيد البعد).
- Nom_Du_Tableau : اسم الجدول وحيد البعد (الشعاع).
- **array** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالشعاع.
- **n** : تمثل عدد قيم أو عناصر الشعاع و التي تمثل بعد الشعاع.
- Type_ Du_Tableau : يمثل نوع الشعاع الذي نريد التصريح به، نوع الشعاع يتحدد بنوع المعطيات التي يحتويها، حيث أن Type_ Du_Tableau تأخذ أحد Les Identificateurs Prédéfinis التالية:

1. ***Integer*** : و التي تعني أن الشعاع المصرح بها من النوع الصحيح، أي أن المعطيات (القيم) التي يحتويها الشعاع من النوع الصحيح.
2. ***Real*** : و التي تعني أن الشعاع المصرح بها من النوع الحقيقي، أي أن المعطيات (القيم) التي يحتويها الشعاع من النوع الحقيقي.
3. ***String*** : و التي تعني أن الشعاع المصرح بها من النوع سلسلة أحرف، أي أن المعطيات (القيم) التي يحتويها الشعاع من النوع سلسلة أحرف.
4. ***Char*** : و التي تعني أن الشعاع المصرح بها من النوع حرف، أي أن المعطيات (القيم) التي يحتويها الشعاع من النوع حرف.

- **;** : فاصلة منقوطة Point Virgule

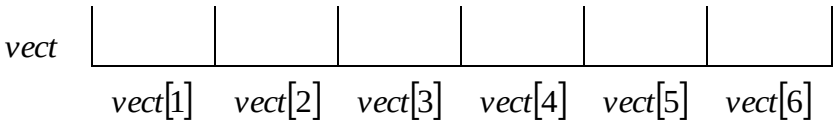
بغرض التوضيح كيفية التصريح بالجداول وحيدة البعد نأخذ المثال التالي:

مثـال 02.10:

1. يتم التصريح بالشعاع من النوع الصحيح و الذي يحمل الاسم *vect* ذو البعد 6 كما يلي:

Var** vect **: array [1..6] of** integer **;

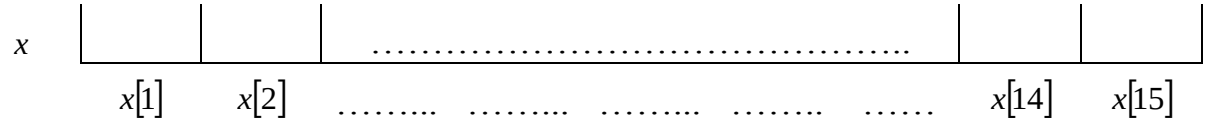
حيث يتم على مستوى الذاكرة تخصيص 6 خانات متجاورة فارغة كما يلي:



2. يتم التصريح بالشعاع من النوع الحقيقي و الذي يحمل الاسم *x* ذو البعد 15 كما يلي:

Var** x **: array [1.. 15] of** real **;

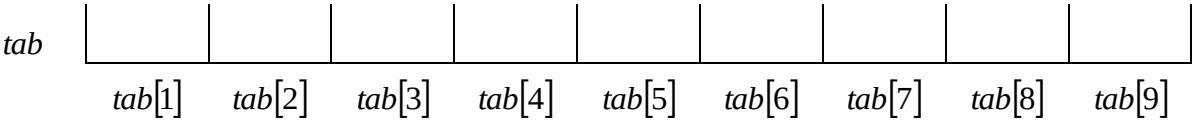
حيث يتم على مستوى الذاكرة تخصيص 15 خانة متجاورة كما يلي:



3. يتم التصريح بالشعاع من النوع سلسلة أحرف و الذي يحمل الاسم *tab* ذو البعد 9 كما يلي:

Var** tab **: array [1.. 9] of** string **;

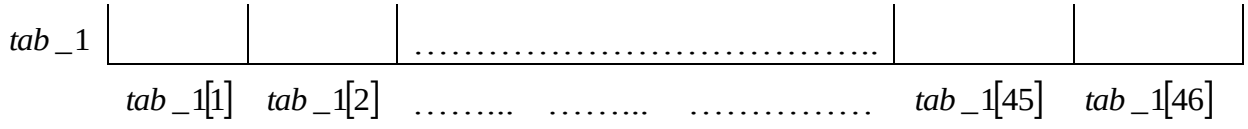
حيث يتم على مستوى الذاكرة تخصيص 9 خانات متجاورة كما يلي:



4. يتم التصريح بالشعاع من النوع حرف و الذي يحمل الاسم *tab_1* ذو البعد 46 كما يلي:

***Var** tab_1 : **array [1..46] of** char ;*

حيث يتم على مستوى الذاكرة تخصيص 46 خانة متجاورة كما يلي:



05.10 – إدخـال المعطيـات فـي الجداول وحيدة (أحادية) البعد (الأشعة):

بعد مرحلة أو عملية التصريح بالجدول وحيد البعد (الشعاع)، ننتقل إلى عملية أو مرحلة إدخال قيم و عناصر (معطيات) هذا الشعاع. عملية إدخال معطيات الشعاع يمكن لها أن تتم بطريقتين[1] هما:

1. إدخال معطيات الشعاع عن طريق البرنامج.
2. إدخال معطيات الشعاع عن لوحة المفاتيح.

سوف يتم التطرق بالتفصيل إلى كل طريقة أدناه:

01.05.10 – إدخـال المعطيـات عن طريـق البرنامـج:

تبعا لهذه الطريقة لإدخال قيم و عناصر (معطيات) الشعاع، فإن هذه المعطيات (مهما كان نوع المتغيرة) تكون موجودة في أو داخل البرنامج أي يتم توجهيها مباشرة في خانات الشعاع الموافقة لها عبر البرنامج مثلما توضحه الأمثلة التالية:

مثـال 03.10:

ليكن الشعاع من النوع الصحيح *tab*1 ذو البعد 7 و الذي يساوي ما يلي:

$$tab1 = \begin{pmatrix} 5 \\ 10 \\ 12 \\ 15 \\ 3 \\ 34 \\ 55 \end{pmatrix}$$

يتم إدخال معطيات الشعاع *tab*1 عن طريق البرنامج كما يلي:

[1] لمزيد من المعلومات حول طرق إدخال المعطيات أنظر الفصل الثالث

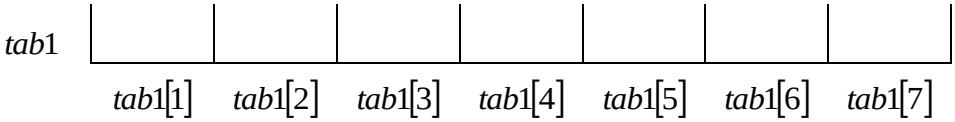
```
tab1[1]:=5;
tab1[2]:=10;
tab1[3]:=12;
tab1[4]:=15;
tab1[5]:=3;
tab1[6]:=34;
tab1[7]:=55;
```

و عليه يصبح البرنامج الذي يسمح بالتصريح بالشعاع *tab*1 و إدخال معطياته هو التالي:

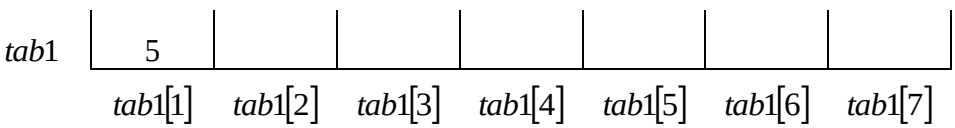
```
program Vecteur_ENTIER ;
var tab1:array[1..7] of integer;
begin
   tab1[1]:=5;
   tab1[2]:=10;
   tab1[3]:=12;
   tab1[4]:=15;
   tab1[5]:=3;
   tab1[6]:=34;
   tab1[7]:=55;
end.
```

يقوم الحاسوب بتنفيذ البرنامج أعلاه كما يلي:

1. عند تنفيذ السطر *var tab1:array[1..7] of integer;* يقوم الحاسوب على مستوى الذاكرة بتخصيص 7 خانات متجاورة كما يلي:



2. عند تنفيذ السطر ***tab1[1]:=5;*** يقوم بتخزين القيمة الصحيحة 5 في الخانة [1]*tab*1 كما هو موضح أدناه:



3. عند تنفيذ السطر ***tab1[2]:=10;*** يقوم بتخزين القيمة الصحيحة 10 في الخانة [1]*tab*1 كما هو موضح أدناه:

*tab*1	5	10					
	*tab*1[1]	*tab*1[2]	*tab*1[3]	*tab*1[4]	*tab*1[5]	*tab*1[6]	*tab*1[7]

4. و هكذا حتى يتم تنفيذ آخر سطر ***tab1[7]:=55;*** لإدخال معطيات الشعاع فيتم الحصول على مستوى الذاكرة على ما يلي:

*tab*1	5	10	12	15	3	34	35
	*tab*1[1]	*tab*1[2]	*tab*1[3]	*tab*1[4]	*tab*1[5]	*tab*1[6]	*tab*1[7]

مثـال 04.10:

ليكن الشعاع من النوع الحقيقي *vect* _1 ذو البعد 5 و الذي يساوي ما يلي:

$tab1 = [12.5 \quad 0.24 \quad 239.879 \quad 3098.0002 \quad 1.6764]$

البرنامج الذي يسمح بإدخال معطيات الشعاع *vect* _1 هو التالي:

```
program Vecteur_REEL ;
var vect_1:array[1..5] of real;
begin
   vect_1[1]:=12.5;
   vect_1[2]:=0.24;
   vect_1[3]:=239.879;
   vect_1[4]:=3098.0002;
   vect_1[5]:=1.6764;
end.
```

مثـال 05.10:

ليكن الشعاع من النوع سلسلة أحرف *nom* ذو البعد 6 و الذي يساوي ما يلي:

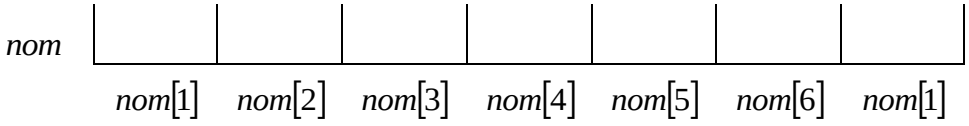
$$nom = \begin{bmatrix} ZAHRA \\ SIHEM \\ IMENE \\ FAIZA \\ DJASMINE \\ KHALDIA \end{bmatrix}$$

البرنامج الذي يسمح بإدخال معطيات الشعاع *nom* هو التالي:

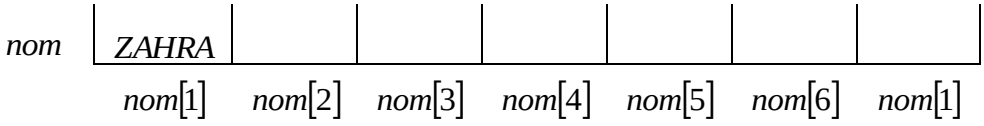
```
program Vecteur_CHAINE_CARACTERES ;
var nom : array[1..6] of string;
begin
   nom [1]:='ZAHRA';
   nom [2]:= 'SIHEM';
   nom [3]:= 'IMENE';
   nom [4]:= 'FAIZA';
   nom [5]:= 'DJASMINE';
   nom [6]:= 'KHALDIA';
end.
```

يقوم الحاسوب بتنفيذ البرنامج أعلاه كما يلي:

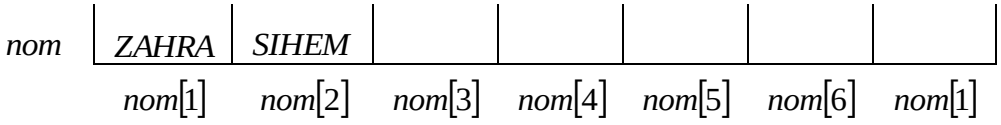
1. عند تنفيذ السطر *var nom : array[1..6] of string;* يقوم الحاسوب على مستوى الذاكرة بتخصيص 6 خانات متجاورة كما يلي:



2. عند تنفيذ السطر **nom [1]:='ZAHRA';** يقوم بتخزين المعطية من النوع سلسلة أحرف *ZAHRA* في الخانة [1]*nom* كما هو موضح أدناه:



3. عند تنفيذ السطر **nom [2]:= 'SIHEM';** يقوم بتخزين المعطية من النوع سلسلة أحرف *SIHEM* في الخانة [2]*nom* كما هو موضح أدناه:



4. و هكذا حتى يتم تنفيذ جميع أسطر إدخال معطيات الشعاع فيتم الحصول على مستوى الذاكرة على ما يلي:

nom	*ZAHRA*	*SIHEM*	*IMENE*	*FAIZA*	*DJASMINE*	*KHALDIA*
	nom[1]	*nom*[2]	*nom*[3]	*nom*[4]	*nom*[5]	*nom*[6]

مثـال 06.10:

ليكن الشعاع من النوع حرف *ch* ذو البعد 5 و الذي يساوي ما يلي:

$$ch = \begin{pmatrix} Z \\ S \\ I \\ F \\ D \end{pmatrix}$$

البرنامج الذي يسمح بإدخال معطيات الشعاع *ch* هو التالي:

```
program Vecteur_CARACTERE ;
var ch : array[1..5] of ch;
begin
   ch [1]:='Z';
   ch [2]:= 'S';
   ch [3]:= 'I';
   ch [4]:= 'F';
   ch [5]:= 'D';
end.
```

يقوم الحاسوب بتنفيذ البرنامج أعلاه كما يلي:

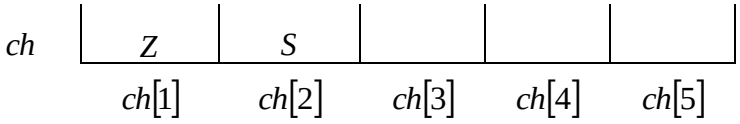
1. عند تنفيذ السطر *var ch : array[1..5] of ch;* يقوم الحاسوب على مستوى الذاكرة بتخصيص 5 خانات متجاورة كما يلي:

ch					
	ch[1]	*ch*[2]	*ch*[3]	*ch*[4]	*ch*[5]

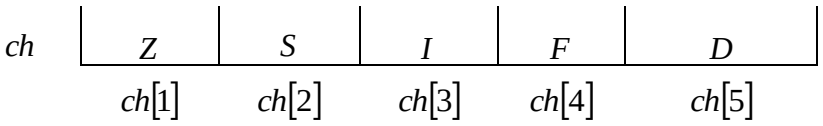
2. عند تنفيذ السطر ***ch [1]:='Z';*** يقوم بتخزين المعطية من النوع حرف *Z* في الخانة *ch*[1] كما هو موضح أدناه:

ch	*Z*				
	ch[1]	*ch*[2]	*ch*[3]	*ch*[4]	*ch*[5]

3. عند تنفيذ السطر ***ch [2]:= 'S';*** يقوم بتخزين المعطية من النوع حرف *S* في الخانة *ch*[2] كما هو موضح أدناه:



4. و هكذا حتى يتم تنفيذ جميع أسطر إدخال معطيات الشعاع فيتم الحصول على مستوى الذاكرة على ما يلي:



02.05.10 – إدخـال المعطيـات عن طريـق لوحـة المفاتيـح:

إضافة إلى طريقة إدخال معطيات الشعاع عن طريق البرنامج و التي مضمونها أن المعطيات تكون موجودة في البرنامج أي يتم توجيهها مباشرة في الخانات الموافقة لها عبر البرنامج، هناك طريقة أخرى لإدخال المعطيات و هي الأكثر استخداما و التي تتمثل في إدخال معطيات الشعاع عن طريق لوحة المفاتيح (Par Le Clavier). مضمون هذه الطريقة هو أنه يتم إدخال معطيات الشعاع بعد تنفيذ البرنامج حيث يطلب الحاسوب من المستخدم إدخال هذه المعطيات ليقوم هذا الأخير بإدخالها عن طريق لوحة المفاتيح. يتم إدخال معطيات الشعاع عن طريق لوحة المفاتيح باستخدام ***Read*** أو ***Readln*** كما توضحه الأمثلة التالية:

مثـال 07.10:

ليكن الشعاع من النوع الصحيح *e* ذو البعد 7 و الذي يساوي ما يلي:

$$e = [13 \quad 0 \quad 157 \quad 2345 \quad 80 \quad 3 \quad 913]$$

يتم إدخال معطيات الشعاع *e* عن طريق لوحة المفاتيح باستخدام ***Read*** أو ***Readln*** كما يلي:

```
Readln ( e[1] ) ;
Readln ( e[2] ) ;
Readln ( e[3] ) ;
Readln ( e[4] ) ;
Readln ( e[5] ) ;
Readln ( e[6] ) ;
Readln ( e[7] ) ;
```

و عليه يصبح البرنامج الذي يسمح بالتصريح بالشعاع *e* و إدخال معطياته هو التالي:

```
program Inroduction_Des_Elements_Du_Vecteur_Par_CLAVIER;
var e:array[1..7] of integer;
begin
    Readln ( e[1] ) ;
    Readln ( e[2] ) ;
    Readln ( e[3] ) ;
    Readln ( e[4] ) ;
    Readln ( e[5] ) ;
    Readln ( e[6] ) ;
    Readln ( e[7] ) ;
end.
```

كما يمكن إدخال عناصر الشعاع *e* على الشكل التالي:

```
Write ('Donnez l'élément Numéro 1 Du Vecteur e :'
Readln ( e[1] ) ;
Write ('Donnez l'élément Numéro 2 Du Vecteur e :'
Readln ( e[2] ) ;
Write ('Donnez l'élément Numéro 3 Du Vecteur e :'
Readln ( e[3] ) ;
Write ('Donnez l'élément Numéro 4 Du Vecteur e :'
Readln ( e[4] ) ;
Write ('Donnez l'élément Numéro 5 Du Vecteur e :'
Readln ( e[5] ) ;
Write ('Donnez l'élément Numéro 6 Du Vecteur e :'
Readln ( e[6] ) ;
Write ('Donnez l'élément Numéro 7 Du Vecteur e :'
Readln ( e[7] ) ;
```

ما يلاحظ أو يقال عن البرنامج الذي يسمح بإدخال عناصر شعاع أنه يحتوي على العديد من الأسطر و بغرض تقليص عدد أسطر البرنامج نقوم باستخدام إحدى الحلقات التي تم التطرق إليها في الفصل السابق سوف نقتصر على استخدام الحلقة *For ... to ... do ...* كما يلي:

الشعاع e ذو البعد 7 له مؤشر و ليكن i يتغير من 1 إلى 7 أي $i = 1,2,3,.....,7$ و عليه يتم استخدام الحلقة ... *For ... to ... do* في إدخال عناصر الشعاع e كما يلي:

```
program Inroduction_Des_Elements_Du_Vecteur_Par_CLAVIER;
var e:array[1..7] of integer;
var i : integer ;
begin
for i:= 1 to 7  do
    Readln ( e[i] ) ;
end.
```

كما يمكن إعادة كتابة البرنامج على الشكل التالي:

```
program Inroduction_Des_Elements_Du_Vecteur_Par_CLAVIER;
var e:array[1..7] of integer;
var i : integer ;
begin
for i:= 1 to 7  do
   begin
      write('Donnez l élément Numéro ', i ,' Du Vecteur e :') ;
      readln ( e[i] ) ;
  end;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

```
Donnez l élément Numéro 1 Du Vecteur e : 13
Donnez l élément Numéro 2 Du Vecteur e : 0
Donnez l élément Numéro 3 Du Vecteur e : 157
Donnez l élément Numéro 4 Du Vecteur e : 2345
Donnez l élément Numéro 5 Du Vecteur e : 80
Donnez l élément Numéro 6 Du Vecteur e : 3
Donnez l élément Numéro 7 Du Vecteur e : 913
```

مثـال 08.10:

ليكن الشعاع من النوع سلسلة أحرف *ville* ذو البعد 10 و الذي يتضمن مجموعة من المدن

$$ville = [TIARET \quad SETIF \quad SKIKDA \quad SAIDA \quad ORAN]$$

يتم إدخال الخمس مدن في الشعاع *ville* عن طريق لوحة المفاتيح باستخدام *Read* أو *Readln* كما يلي:

```
program Inroduction_Des_Elements_Du_Vecteur_Par_CLAVIER;
var ville :array[1..5] of string;
var i : integer ;
begin
for i:= 1 to 5  do
   begin
      write('Donnez Le Nom De La Ville Numéro ', i ) ;
      readln ( ville[i] ) ;
  end;
end.
```

06.10 – العمليـات علـى الجداول وحيدة (أحادية) البعد (الأشعة):

هناك العديد من العمليات الممكن إجراءها على الأشعة من نفس البعد نذكر على سبيل المثال لا الحصر العمليات التالية:

1. جمع شعاعين.
2. فرق شعاعين
3. جداء شعاعين.
4. حساب مجموع عناصر شعاع معين.
5. حساب المتوسط الحسابي لعناصر شعاع معين.
6. حساب الانحراف المعياري لعناصر شعاع معين.

من خلال هذه الفقرة سوف يتم التطرق إلى بعض هذه العمليات

01.06.10 – جمـع شعاعيـن:

ليكن الشعاعين A و B من النوع الصحيح و لهما نفس البعد 8 حيث أن:

$$A=\begin{pmatrix}15\\34\\6\\7\\80\\0\\45\\5\end{pmatrix} \quad , \quad B=\begin{pmatrix}4\\6\\12\\0\\38\\21\\78\\8\end{pmatrix}$$

جمع الشعاعين A و B عبارة عن شعاع ثالث و ليكن C له نفس بعدهما و الذي يساوي 8 حيث أن:

$$C = A + B$$

بغرض كتابة البرنامج الذي يسمح بجمع الشعاعين A و B في الشعاع C نتبع الخطوات التالية:

1. التصريح بالشعاع A ذو البعد 8 من النوع الصحيح: يتم ذالك كما يلي

var a : array[1..8] of integer ;

2. التصريح بالشعاع B ذو البعد 8 من النوع الصحيح: يتم ذالك كما يلي

var b : array[1..8] of integer ;

3. التصريح بالشعاع C ذو البعد 8 من النوع الصحيح: يتم فيه تخزين نتيجة جمع عناصر الشعاعين A و B

var c : array[1..8] of integer ;

4. التصريح بالمتغيرة i من النوع الصحيح و التي تستخدم كمؤشر للأشعة الثلاثة

var i : integer ;

5. إدخال معطيات أو عناصر الشعاع A : يتم ذالك كما يلي

```
for i:= 1 to 8  do
   begin
      write('Donnez l élément Numéro ', i ,' Du Vecteur A :') ;
      readln ( a[i] ) ;
   end;
```

6. إدخال معطيات أو عناصر الشعاع B : يتم ذالك كما يلي

```
for i:= 1 to 8  do
   begin
      write('Donnez l élément Numéro ', i ,' Du Vecteur B :') ;
      readln ( b[i] ) ;
```

```
end;
```

7. حساب قيم الشعاع C كما يلي:

```
for i:= 1 to 8  do
   c [i] := a [i] +  b[i] ;
```

8. نشر عناصر الشعاع C كما يلي:

```
for i:= 1 to 8  do
   writeln ( c [i] ) ;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program SOMME_DE_DEUX_VECTEURS;
var a : array[1..8] of integer ;
   b : array[1..8] of integer ;
   c : array[1..8] of integer ;
   i : integer ;
begin
{Introduction Du Vecteur A}
for i:= 1 to 8  do
   begin
    write('Donnez l élément Numéro ', i ,' Du Vecteur A :') ;
    readln ( a[i] ) ;
   end;
{Introduction Du Vecteur B}
for i:= 1 to 8  do
   begin
    write('Donnez l élément Numéro ', i ,' Du Vecteur B :') ;
    readln ( b[i] ) ;
   end;
{Calcul Du Vecteur C}
for i:= 1 to 8  do
    c [i] := a [i] +  b[i]  ;
```

```
{Affichage Du Vecteur C}
for i:= 1 to 8  do
    writeln ( c [i] ) ;
readln ;
end.
```

يمكن التقليص من عدد أسطر البرنامج عن طريق استخدام حلقة واحدة كما يلي:

```
program SOMME_DE_DEUX_VECTEURS;
var a : array[1..8] of integer ;
   b : array[1..8] of integer ;
   c : array[1..8] of integer ;
   i : integer ;
begin
for i:= 1 to 8  do
   begin
     {Introduction Du Vecteur A}
     write('Donnez l élément Numéro ', i ,' Du Vecteur A :') ;
     readln ( a[i] ) ;
     {Introduction Du Vecteur B}
     write('Donnez l élément Numéro ', i ,' Du Vecteur B :') ;
     readln ( b[i] ) ;
     {Calcul Du Vecteur C}
     c [i] := a [i] +  b[i]  ;
  end;
{Affichage Du Vecteur C}
for i:= 1 to 8  do
    writeln ( c [i] ) ;
readln ;
end.
```

02.06.10 – فـرق شعاعيـن:

البرنامج الذي يسمح بحساب فرق شعاعين A و B هو نفسه الذي يسمح بحساب مجموعهما الذي يتغير هو السطر التالي

```
c [i] := a [i] + b[i] ;
```

و الذي يتم استبداله بالسطر التالي:

```
c [i] := a [i] - b[i] ;
```

و عليه يصبح البرنامج هو التالي:

```
program DIFFERENCE_DE_DEUX_VECTEURS;
var a : array[1..8] of integer ;
   b : array[1..8] of integer ;
   c : array[1..8] of integer ;
   i : integer ;
begin
{Introduction Du Vecteur A}
for i:= 1 to 8  do
   begin
    write('Donnez l élément Numéro ', i ,' Du Vecteur A :') ;
    readln ( a[i] ) ;
   end;
{Introduction Du Vecteur B}
for i:= 1 to 8  do
   begin
    write('Donnez l élément Numéro ', i ,' Du Vecteur B :') ;
    readln ( b[i] ) ;
   end;
{Calcul Du Vecteur C}
for i:= 1 to 8  do
    c [i] := a [i] - b[i]  ;
{Affichage Du Vecteur C}
for i:= 1 to 8  do
```

```
    writeln ( c [i] ) ;
readln ;
end.
```

03.06.10 – جداء شعاعين:

كما هو معلوم نتيجة جداء شعاعين عبارة عن عدد و ليس شعاع، و عليه فإن جداء الشعاعين A و B و الذي يرمز له بالرمز $A\times B$ يساوي ما يلي:

$$A\times B=\begin{pmatrix}15 & 34 & 6 & 7 & 80 & 0 & 45 & 5\end{pmatrix}\times\begin{pmatrix}4\\6\\12\\0\\38\\21\\78\\8\end{pmatrix}$$

$$A\times B=15\times 4+34\times 6+6\times 12+7\times 0+80\times 38+0\times 21+45\times 78+5\times 8$$

بغرض كتابة البرنامج الذي يسمح بحساب $A\times B$ نتبع الخطوات التالية:

1. التصريح بالشعاع A ذو البعد 8 من النوع الصحيح:
2. التصريح بالشعاع B ذو البعد 8 من النوع الصحيح:
3. التصريح بالمتغيرة i من النوع الصحيح و التي تستخدم كمؤشر للشعاعين
4. التصريح بالمتغيرة s التي يتم فيها تخزين نتيجة جداء الشعاعين أو مجموع الجداء عناصر الشعاعين

```
s : integer ;
```

5. إدخال عناصر الشعاع A:
6. إدخال عناصر الشعاع B:
7. حساب الجداء $A\times B$ كما يلي:

```
for i:= 1 to 8  do
  s := s + a [i] *  b[i] ;
```

8. نشر نتيجة الجداء المخزنة في المتغيرة s كما يلي:

```
writeln ( 'Le Produit  A*B = ' , s ) ;
```

و عليه يصبح البرنامج هو التالي:

```
program PRODUIT_DE_DEUX_VECTEURS;
var a : array[1..8] of integer ;
   b : array[1..8] of integer ;
   i , s : integer ;
begin
{Introduction Du Vecteur A}
for i:= 1 to 8  do
   begin
    write('Donnez l élément Numéro ', i ,' Du Vecteur A :') ;
    readln ( a[i] ) ;
   end;
{Introduction Du Vecteur B}
for i:= 1 to 8  do
   begin
    write('Donnez l élément Numéro ', i ,' Du Vecteur B :') ;
    readln ( b[i] ) ;
   end;
{Calcul Du Produit A*B}
for i:= 1 to 8  do
    s:= s + a [i] *  b[i]  ;
{Affichage Du Résultat De A*B}
writeln ( 'Le Produit A*B = ',s ) ;
readln ;
end.
```

07.10 – فـرز الجداول وحيدة (أحادية) البعد (الأشعة):

نعني بعملية فرز جدول ترتيب عناصر هذا الجدول تنازليا أو تصاعديا إضافة إلى البحث عن أكبر قيمة أو أقل قيمة يتضمنها الجدول أو الشعاع

01.07.10 – البحث عن أكبر قيمـة:

سوف يتم التطرق من خلال هذه الفقرة إلى كيفية إيجاد أكبر عنصر من بين عناصر شعاع من خلال المثال التالي:

مثـال 09.10:

لنفرض أننا نتوفر على شعاع ليكن x ذو البعد 5 و الذي يساوي ما يلي:

$$x = [25 \quad 45 \quad 10 \quad 50 \quad 5]$$

حيث نقوم في هذا الشعاع بالبحث عن أكبر قيمة و لنسميها max من بين القيم التي يحتويها

بغرض إيجاد أكبر قيمة max نقترح الطريقة التالية[1]:

- نقوم بافتراض أن أكبر قيمة تساوي القيمة أو العنصر الأول في الجدول أي: $\max := x[1]$
- نقوم بمقارنة القيمة max مع جميع القيم المتبقية (الأخرى) للشعاع x قيمة بعد قيمة أي مع القيم $x[2], x[3], x[4]x[5]$

حيث:

إذا كانت القيمة max أقل من القيمة $x[i]$ أي $\max < x[i]$ **فإنه** يتم استبدال القيمة max بالقيمة $x[i]$ أي $\max := x[i]$ **و إلا** فإنه لا يتم إجراء أي استبدال و إنما يتم المرور إلى القيمة الموالية للجدول بغرض مقارنتها بالقيمة max عملية المقارنة هذه تتم كما يلي:

```
For i:=1 to 5 do
    If max < x[i] then max := x[i] ;
```

و عليه يصبح البرنامج هو التالي:

```
program Le_MAXIMUM_Du_VECTEUR;
var x : array[1..5] of integer ;
  i , max : integer ;
begin
{Introduction Du Vecteur X}
for i:= 1 to 5  do
  begin
   write('Donnez l élément Numéro ', i ,' Du Vecteur X :') ;
   readln ( x[i] ) ;
  end;
{Comparaison max Avec Le Reste Des élements }
```

[1] نشير إلى أن هذه الطريقة المقترحة ليست الوحيدة بل هناك طرق أخرى لإيجاد أكبر قيمة

```
max:=x[1];
For i:=2 to 5 do
   If max < x[i] then max := x[i] ;
{Affichage De La Valeur max}
writeln ( 'Le MAX = ',max ) ;
readln ;
end.
```

02.07.10 – البحث عن أقل قيمـة:

سوف يتم التطرق من خلال هذه الفقرة إلى كيفية إيجاد أقل عنصر من بين عناصر شعاع من خلال المثال التالي:

مثـال 10.10:

لنفرض أننا نتوفر على شعاع ليكن y ذو البعد 9 و الذي يساوي ما يلي:

$$y = [3 \quad 23 \quad 67 \quad 99 \quad 33 \quad 7 \quad 0 \quad 87 \quad 309]$$

حيث نقوم في هذا الشعاع بالبحث عن أقل قيمةو لنسميها min من بين القيم التي يحتويها بغرض إيجاد أقل قيمة min نقترح الطريقة التالية:

- نقوم بافتراض أن أقل قيمة تساوي القيمة أو العنصر الأول في الجدول أي: $\min := y[1]$
- نقوم بمقارنة القيمة min مع جميع القيم المتبقية (الأخرى) للشعاع x قيمة بعد قيمة أي مع القيم $y[2], y[3], y[4]y[5]$.....,$y[9]$

حيث:

إذا كانت القيمة min أكبر من القيمة $y[i]$ أي $\min < y[i]$ **فإنه** يتم استبدال القيمة min بالقيمة $x[i]$ أي $\min := y[i]$ **و إلا** فإنه لا يتم إجراء أي استبدال و إنما يتم المرور إلى القيمة الموالية للجدول بغرض مقارنتها بالقيمة min عملية المقارنة هذه تتم كما يلي:

```
For i:=1 to 9 do
   If min > y[i] then min := y[i] ;
```

و عليه يصبح البرنامج هو التالي:

```
program Le_MINIMUM_Du_VECTEUR;
var y : array[1..9] of integer ;
  i , min : integer ;
begin
```

```
{Introduction Du Vecteur X}
for i:= 1 to 9  do
   begin
    write('Donnez l élément Numéro ', i ,' Du Vecteur Y :') ;
    readln ( y[i] ) ;
   end;
{Comparaison minAvec Le Reste Des élements }
min:=y[1];
For i:=2 to 9 do
    If min >y[i] then min := y[i] ;
{Affichage De La Valeur min}
writeln ( 'Le MIN = ',min ) ;
readln ;
end.
```

تمارين الفصـــل

✍ التمـريـن 01.10:

1. قدم التعليمة التي تسمح بالتصريح (Declaration) بجدول وحيد البعد (شعاع) Tableau Unidimensionnel من النوع **الصحيح** (Entier).
2. قدم طريقة كتابة (La Syntaxe) التعليمة المتوصل إليها في السؤال الأول.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول وحيد البعد (شعاع) من النوع **الصحيح** ذو البعد 25.

✍ التمـريـن 02.10:

1. قدم التعليمة التي تسمح بالتصريح (Declaration) بجدول وحيد البعد (شعاع) Tableau Unidimensionnel من النوع **الحقيقي** صحيح (Réel).
2. قدم طريقة كتابة (La Syntaxe) التعليمة المتوصل إليها في السؤال الأول.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول وحيد البعد (شعاع) من النوع **الحقيقي** ذو البعد 46.

✍ التمـريـن 03.10:

1. قدم التعليمة التي تسمح بالتصريح (Declarations) بجدول وحيد البعد (شعاع) Tableau Unidimensionnel من النوع **أحرف** (caractères).
2. قدم طريقة كتابة (La Syntaxe) التعليمة المتوصل إليها في السؤال الأول.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول وحيد البعد (شعاع) من النوع **أحرف** ذو البعد 38.

✍ التمـريـن 04.10:

1. قدم التعليمة التي تسمح بالتصريح (Declaration) بجدول وحيد البعد (شعاع) Tableau Unidimensionnel من النوع **سلسلة أحرف** (Chaine de caractères).
2. قدم طريقة كتابة (La Syntaxe) التعليمة المتوصل إليها في السؤال الأول.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول وحيد البعد (شعاع) من النوع **سلسلة أحرف** ذو البعد 100.

✍ التمـريـن 05.10:

لتكن القيم أو العناصر التالية:

15 30 41 44 100 23 35 46 39 50 56

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال العناصر أعلاه عن طريق البرنامج (التوجيه المباشر) L'affèctation directe في جدول وحيد البعد (شعاع) الذي يحمل الإسم Tab. أي أن العناصر تكون ضمن البرنامج.

✍ التمــريــن 06.10:

لتكن معطيات التمرين 01.10 أعلاه

- **المطلـوب:**

1. باستخدام التعليمة **for := to do** أكتب البرنامج الذي يسمح بإدخال العناصر أعلاه عن طريق لوحة المفاتيح الملامس (Le clavier) في جدول وحيد البعد (شعاع).

✍ التمــريــن 07.10:

لتكن معطيات التمرين 01.10 أعلاه

- **المطلـوب:**

1.باستخدام التعليمة **while….do….** أكتب البرنامج الذي يسمح بإدخال العناصر أعلاه عن طريق لوحة المفاتيح الملامس (Le clavier) في جدول وحيد البعد (شعاع).

✍ التمــريــن 08.10:

لتكن معطيات التمرين 01.10 أعلاه

- **المطلـوب:**

1.باستخدام التعليمة **repeat …. until ….** أكتب البرنامج الذي يسمح بإدخال العناصر أعلاه عن طريق لوحة المفاتيح الملامس (Le clavier) في جدول وحيد البعد (شعاع).

✍ التمــريــن 09.10:

لتكن معطيات التمرين 01.10 أعلاه

- **المطلـوب:**

أكتب البرنامج الذي يسمح بنشر عناصر الجدول Tab المتوصل إليه في التمرين 01.10 أعلاه كما يلي:

1. الحالة الأولى:

56 50 39 46 35 23 100 44 41 30 15

2. الحالة الثانية:

15 30 41 44 100 23 35 46 39 50 56

3. الحالة الثالثة:

15
30
41
44
100
23
35
46
39
50
56

4. الحالة الرابعة:

56
50
39
46
35
23
100
44
41
30
15

✍ التمــريــن 10.10:

ليكن لدينا الشعاعين $Vect1$ و $Vect2$ التاليين:

$Vect1$:	5	15	25	35	45	.	.	.	.	.	75	85	95

$Vect2$:	10	20	30	40	50	.	.	.	.	.	80	90	100

- **المطلــوب:**

1. أكتب البرنامج الذي يسمح بإدخال عناصر الشعاع الأول $Vect1$.
2. أكتب البرنامج الذي يسمح بإدخال عناصر الشعاع الأول $Vect2$.
3. أكتب البرنامج الذي يسمح بدمج (Fusion) الشعاعين $Vect1$ و $Vect2$ في شعاع ثالث $Vect3$ كما يلي:

$Vect3$:	5	10	15	20	25	30	35	40	.	.	.	.	90	95	100

✍ التمــريــن 11.10:

ليكن الشعاع $Vect3$ المتوصل إليه في التمرين 10.10 أعلاه

- **المطلــوب:**

1. أكتب البرنامج الذي يسمح بتقسيم الشعاع $Vect3$ إلى الشعاعين $Vect1$ و $Vect2$ المقدمين من خلال التمرين 01.10 أعلاه.

✍ التمــريــن 12.10:

القيم أو المعطيات أدناه تمثل علامات الطالب XXX في 12 مقياس

12.5 7.75 8 14.25 13 12 7.5 10.5 11.25 15.25 11 9.75

- **المطلــوب:**

1. أكتب البرنامج الذي يسمح بإدخال علامات الطالب في شعاع نسميه Note.
2. أكتب البرنامج الذي يسمح بحساب معدل الطالب و نشره على الشاشة كما يلي:

La Moyenne De L'étudiant XXX est :

نشير إلى أن معدل الطالب يساوي إلى مجموع العلامات مقسوما على عددها

✍ التمــريــن 13.10:

لتكن العلامات المقدمة من خلال التمرين 03.10 أعلاه و لتكن معاملاتها التالية:

2 3 1 1 3 2 2 1 3 3 3 1

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال معاملات علامات الطالب في شعاع نسميه Coe.

2. أكتب البرنامج الذي يسمح بحساب معدل الطالب و نشره على الشاشة كما يلي:

La Moyenne De L'étudiant XXX est :

نشير إلى أن معدل الطالب يساوي إلى مجموع جداء العلامات بمعاملاتها مقسوما على مجموع المعاملات.

✍ التمــريــن 14.10:

ليكن لدينا الشعاعين A و B لهما نفس البعد

A :

B :

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال عناصر الشعاع الأول A.

2. أكتب البرنامج الذي يسمح بإدخال عناصر الشعاع الثاني B.

3. أكتب البرنامج الذي يسمح بحساب و نشر عناصر الشعاع الثالث C و الذي يمثل مجموع الشعاعين A و B أي $C = A + B$.

4. أكتب البرنامج الذي يسمح بحساب و نشر عناصر الشعاع الثالث D و الذي يمثل مربع عناصر الشعاع A.

5. أكتب البرنامج الذي يسمح بحساب و نشر عناصر الشعاع الثالث E و الذي يمثل مربع عناصر الشعاع B.

✍ التمــريــن 15.10:

لتكن ألقاب و أسماء مجموعة من الطلبة التالية:

YA MENAD KHALIL MAZOUZI YAMINI HALFAOUI

ZAHRA SIHEM DJASMINE FAIZA IMENE LAMIA

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال الألقاب أعلاه عن طريق البرنامج (التوجيه المباشر) L'affèctation directe في جدول وحيد البعد (شعاع) الذي يحمل الاسم NOM. أي أن العناصر تكون ضمن البرنامج.

2. إدخال الأسماء أعلاه عن طريق البرنامج (التوجيه المباشر) L'affèctation directe في جدول وحيد البعد (شعاع) الذي يحمل الاسم PRENOM. أي أن العناصر تكون ضمن البرنامج.

✍ التمـريـن 16.10:

لتكن معطيات التمرين 15.10 أعلاه

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال الألقاب أعلاه عن طريق لوحة المفاتيح Le clavier في جدول وحيد البعد (شعاع) *NOM* .

2. إدخال الأسماء أعلاه عن طريق لوحة المفاتيح Le clavier في جدول وحيد البعد (شعاع) *PRENOM* .

حيث أن عملية إدخال المعطيات (الاسم و اللقب) تتم على الشكل التالي:

1- / <u>Introduction Des Noms Des Etudiants :</u>

Donnez Le Nom De L'étudiant Numéro 1 :
Donnez Le Nom De L'étudiant Numéro 2 :
Donnez Le Nom De L'étudiant Numéro 3 :
Donnez Le Nom De L'étudiant Numéro 4 :
Donnez Le Nom De L'étudiant Numéro 5 :
Donnez Le Nom De L'étudiant Numéro 6 :

2- / <u>Introduction Des Prénoms Des Etudiants :</u>

Donnez Le Prénom De L'étudiant Numéro 1 :
Donnez Le Prénom De L'étudiant Numéro 2 :
Donnez Le Prénom De L'étudiant Numéro 3 :
Donnez Le Prénom De L'étudiant Numéro 4 :
Donnez Le Prénom De L'étudiant Numéro 5 :
Donnez Le Prénom De L'étudiant Numéro 6 :

التمــريــن 17.10:

لتكن معطيات التمرين 15.10 أعلاه

- **المطلـوب:**

أكتب البرنامج الذي يسمح بنشر عناصر الشعاعين *NOM* و *PRENOM* المتوصل إليهما في التمرين 15.10 أعلاه كما يلي:

1. الحالة الأولى:

YA MENAD KHALIL MAZOUZI YAMINI HALFAOUI

ZAHRA SIHEM DJASMINE FAIZA IMENE LAMIA

2. الحالة الثانية:

HALFAOUI YAMINI MAZOUZI KHALIL MENAD YA

LAMIA IMENE FAIZA DJASMINE SIHEM ZAHRA

3. الحالة الثالثة:

NOM	PRENOM
YA	ZAHRA
MENAD	SIHEM
KHALIL	DJASMINE
MAZOUZI	FAIZA
YAMINI	IMENE
HALFAOUI	LAMIA

4. الحالة الرابعة:

NOM	PRENOM
HALFAOUI	LAMIA
YAMINI	IMENE
MAZOUZI	FAIZA

KHALIL DJASMINE
MENAD SIHEM
YA ZAHRA

✍ التمــريــن 17.10:

لتكن الأسماء التالية:

ZAH SIH DJA FAI IME LAM SOU NAI HAM

و التي تحصلت على المعدلات التالية:

16.45 14.34 14.04 10.12 14.00 16.03 12.76 15 11.07

- **المطلــوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال الأسماء جدول وحيد البعد (شعاع) الذي يحمل الإسم Nom
2. إدخال المعدلات جدول وحيد البعد (شعاع) الذي يحمل الإسم Moy
3. نشر الأسماء و المعدلات على الشكل التالي:

NOM	MOYENNE
ZAH	16.45
SIH	14.34
DJA	14.04
FAI	10.12
IME	14.00
LAM	16.03
SOU	12.76
NAI	15
HAM	11.07

✍ التمــريــن 18.10:

لتكن القيم التالية:

23 10 45 78 120 250 350 45 301 25 78 89 90 98 101

- **المطلــوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال القيم أعلاه في جدول وحيد البعد (شعاع) X .
2. حساب مجموع قيم الشعاع X .
3. حساب متوسط قيم الشعاع X .
4. نشر قيم الشعاع X .
5. نشر مجموع قيم الشعاع X .
6. نشر متوسط قيم الشعاع X .

✍ التمـريـن 19.10:

ليكن الشعاع X المتوصل إليه في التمرين 01.10 أعلاه

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. حساب قيم الجدول وحيد البعد (الشعاع) Y و التي تمثل مربع قيم الشعاع X .
2. حساب متوسط قيم الشعاع Y .
3. حساب متوسط قيم الشعاع Y .
4. نشر قيم الشعاع Y .
5. نشر مجموع قيم الشعاع Y .
6. نشر متوسط قيم الشعاع Y .

✍ التمـريـن 20.10:

أكتب البرنامج الذي يسمح بما يلي:

1. إدخال 200 رمز منتج في شعاع *CODE _ PRODUIT* .
2. إدخال 200 اسم منتج في شعاع *NOM _ PRODUIT* .
3. إدخال سعر الوحدة الواحدة من كل منتج في شعاع *PRIX _ UNITE* .
4. إدخال الكمية المتاحة من كل منتج في شعاع *QUANT _ DISPO* .
5. نشر ما يلي:

Code Produit	Nom Produit	Quantité Disponible	Prix
1	SUCRE	50	100
2	CAFE	91	150
3	THE	34	100
4	TOMATE	69	200

.	.	.	.
.	.	.	.
.	.	.	.
200	.	.	.

✍ التمـريــن 21.10:

انطلاقا من معطيات التمرين 20.10 أعلاه

أكتب البرنامج الذي يسمح بما يلي:

1. إدخال الكمية المطلوبة من كل منتج في شعاع *QUANT _ DEMAN* .
2. حساب قيمة كل منتج في شعاع *VALEURS _ PRODUIT* . حيث أن قيمة كل منتج تساوي السعر مضروب بالكمية المطلوبة أي:

$$VALEURS_PRODUIT = PRIX_UNITE \;\; X \;\; QUANT_DEMAN$$

3. حساب الكمية المتبقية من كل منتج في شعاع *QUANT _ RESTAN* . حيث أن قيمة الكمية المتبقية من كل منتج تساوي الكمية المتاحة مطروحا منها الكمية المطلوبة أي:

$$QUANT_RESTAN = QUANT_DISPO \; - \; QUANT_DEMAN$$

4. نشر الفاتورة التالية:

Code Produit	Nom Produit	Quantité Demandée	Prix	Valeur
4	TOMATE	8	200	1600
1	SUCRE	3	100	300
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
2	CAFE	50	5	250
	T O T A L F A C T U R E			……..

5. نشر ما يلي:

Code Produit	Nom Produit	Quantité Restante
1	SUCRE	47

2	CAFE	41
3	THE	34
4	TOMATE	61
.	.	.
.	.	.
.	.	.
200	.	.

✍ التمـريـن 22.10:

أكتب البرنامج الذي يسمح بما يلي:

1. إدخال ألقاب مجموعة من الطلبة و عددهم 40 في شعاع *NOM*.
2. إدخال أسماء مجموعة الطلبة في شعاع *PRENOM*.
3. إدخال علامات مجموعة الطلبة لخمسة مقاييس في خمسة أشعة هي:

$MODULE-1$, $MODULE-2$, $MODULE-3$, $MODULE-4$, $MODULE-5$

4. إدخال معاملات الخمسة مقاييس في شعاع *COEF*.
5. حساب معدلات الطلبة في شعاع *MOYENNE*. نشير إلى أن معدل الطالب يساوي إلى مجموع جداء العلامات بمعاملاتها مقسوما على مجموع المعاملات.
6. حساب نتائج (RESULTATS) الطلبة في شعاع *RESULTATS*. حيث أنه إذا كان معدل الطالب يساوي أو يفوق 10 فإن الطالب يعد ناجحا (ADMIS) أما إذا كان معدل الطالب يقل عن 10 فإن الطالب يعد راسبا (AJOURNE).
7. نشر النتائج كما يلي:

NOM	PRENOM	MOD1	MOD2	MOD3	MOD4	MOD5	MOY	RESULTATS
SETTI	HAMID	10.50	11.75	08	16.25	13.50		ADMIS
SEDDI	ALI	5.25	03	10.75	12	02		AJOURNE
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.

.	.	.	.	.	.	.	.	.
MAHI	AEK	00.75	03	4.5	13	07	4.75	AJOURNE

✍ التمــريــن 23.10:

المتغير الإحصائي X يأخذ القيم التالية:

X_i
100
35.5
45
06
49
50.2
78
87.3
39
200
55
31

- **المطلــوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال قيم المتغير الإحصائي X في جدول وحيد البعد (شعاع) X.
2. حساب مجموع قيم المتغير الإحصائي X.
3. حساب المتوسط الحسابي $\overline{X}$ للمتغير الإحصائي X. حيث أن:

$$\overline{X} = \frac{\sum_{i=1}^{n=12} X_i}{n=12} = \frac{X_1 + X_2 + X_3 + ... + X_{11} + X_{12}}{n=12}$$

4. حساب مربع قيم المتغير الإحصائي X و إدخالها في شعاع X_CARRE.
5. حساب انحرافات قيم المتغير الإحصائي X عن الوسط الحسابي $\overline{X}$ و إدخالها في شعاع $ECART$. حيث أن:

$$ECART = \left(X_i - \overline{X}\right)$$

6. حساب مربع انحرافات قيم المتغير الإحصائي X عن الوسط الحسابي $\overline{X}$ و إدخالها في شعاع $ECART_CARRE$. حيث أن

$$ECART_CARRE = (ECART)^2 = (X_i - \overline{X})^2$$

7. حساب التباين $Var(X)$ للمتغير الإحصائي X . حيث أن:

$$Var(X) = \frac{\sum_{i=1}^{n=12}(X_i - \overline{X})^2}{n=12} = \frac{(X_1 - \overline{X})^2 + (X_2 - \overline{X})^2 + (X_3 - \overline{X})^2 + ... + (X_{12} - \overline{X})^2}{n=12}$$

8. حساب العزم البسيط من الدرجة الثانية M_S^2 للمتغير الإحصائي X . حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 2 : M_S^2 = \frac{\sum_{i=1}^{n=12} X_i^2}{n=12} = \frac{X_1^2 + X_2^2 + X_3^2 + ... + X_{11}^2 + X_{12}^2}{n=12}$$

9. حساب العزم البسيط من الدرجة الثالثة M_S^3 للمتغير الإحصائي X . حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 3 : M_S^3 = \frac{\sum_{i=1}^{n=12} X_i^3}{n=12} = \frac{X_1^3 + X_2^3 + X_3^3 + ... + X_{11}^3 + X_{12}^3}{n=12}$$

10. حساب العزم البسيط من الدرجة الرابعة M_S^4 للمتغير الإحصائي X . حيث أن:

$$Moment\ \ Simple\ \ D'ordre\ \ 4 : M_S^4 = \frac{\sum_{i=1}^{n=12} X_i^4}{n=12} = \frac{X_1^4 + X_2^4 + X_3^4 + ... + X_{11}^4 + X_{12}^4}{n=12}$$

11. حساب العزم المركزي من الدرجة الأولى M_C^1 للمتغير الإحصائي X . حيث أن:

$$M_C^1 = \frac{\sum_{i=1}^{n=12}(X_i - \overline{X})^1}{n=12} = \frac{(X_1 - \overline{X})^1 + (X_2 - \overline{X})^1 + (X_3 - \overline{X})^1 + ... + (X_{12} - \overline{X})^1}{n=12}$$

12. حساب العزم المركزي من الدرجة الثانية M_C^2 للمتغير الإحصائي X . حيث أن:

$$M_C^2 = \frac{\sum_{i=1}^{n=12}(X_i - \overline{X})^2}{n=12} = \frac{(X_1 - \overline{X})^2 + (X_2 - \overline{X})^2 + (X_3 - \overline{X})^2 + ... + (X_{12} - \overline{X})^2}{n=12}$$

13. حساب العزم المركزي من الدرجة الثالثة M_C^3 للمتغير الإحصائي X . حيث أن:

$$M_C^3 = \frac{\sum_{i=1}^{n=12}(X_i - \overline{X})^3}{n=12} = \frac{(X_1 - \overline{X})^3 + (X_2 - \overline{X})^3 + (X_3 - \overline{X})^3 + ... + (X_{12} - \overline{X})^3}{n=12}$$

14. نشر النتائج كما يلي:

	X	X^2	X^3	$(X - \overline{X})$	$(X - \overline{X})^2$
1	100	.	.	.	.

2	35.5	.	.	.	.
3	45	.	.	.	.
4	06	.	.	.	.
5	49	.	.	.	.
6	50.2	.	.	.	.
7	78	.	.	.	.
8	87.3	.	.	.	.
9	39	.	.	.	.
10	200	.	.	.	.
11	55	.	.	.	.
12	31	.	.	.	.
Total	.	.	.	.	.

- La Somme Des Valeurs De X : .....
- La Moyènne Arithmétique $\overline{X}$: .....
- La Variance $Var(X)$: .....
- Le Moment Simple D'ordre 2 M_S^2 : .....
- Le Moment Simple D'ordre 3 M_S^3 : .....
- Le Moment Simple D'ordre 4 M_S^4 : .....
- Le Moment Centré D'ordre 1 M_C^1 : .....
- Le Moment Centré D'ordre 2 M_C^2 : .....
- Le Moment Centré D'ordre 3 M_C^3 : .....

✍ التمـريـن 24.10:

لنفرض أن المتغير الإحصائي X يأخذ نفس القيم التي يأخذها في التمرين 23.10 بالتكرارات n_i التالية:

X_i	n_i
100	4
35.5	1
45	3

06	2
49	2
50.2	1
78	3
87.3	4
39	5
200	5
55	2
31	3

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

نفس أسئلة التمرين 23.10 حيث أن:

- $\overline{X} = \dfrac{\sum_{i=1}^{12} X_i \times n_i}{\sum_{i=1}^{12} n_i} = \dfrac{X_1 \times n_1 + X_2 \times n_2 + X_3 \times n_3 + ... + X_{11} \times n_{11} + X_{12} \times n_{12}}{n_1 + n_2 + n_3 + ... + n_{11} + n_{12}}$

- $Var(X) = \dfrac{\sum_{i=1}^{n=12} \left(X_i - \overline{X}\right)^2 \times n_i}{\sum_{i=1}^{12} n_i} = \dfrac{\left(X_1 - \overline{X}\right)^2 \times n_1 + \left(X_2 - \overline{X}\right)^2 \times n_2 + \left(X_3 - \overline{X}\right)^2 \times n_3 + ... + \left(X_{12} - \overline{X}\right)^2 \times n_{12}}{n_1 + n_2 + n_3 + ... + n_{11} + n_{12}}$

- $M_S^2 = \dfrac{\sum_{i=1}^{n=12} X_i^{\,2} \times n_i}{\sum_{i=1}^{12} n_i} = \dfrac{X_1^{\,2} \times n_1 + X_2^{\,2} \times n_2 + X_3^{\,2} \times n_3 + ... + X_{11}^{\,2} \times n_{11} + X_{12}^{\,2} \times n_{12}}{n_1 + n_2 + n_3 + ... + n_{11} + n_{12}}$

- $M_S^3 = \dfrac{\sum_{i=1}^{n=12} X_i^{\,3} \times n_i}{\sum_{i=1}^{12} n_i} = \dfrac{X_1^{\,3} \times n_1 + X_2^{\,3} \times n_2 + X_3^{\,3} \times n_3 + ... + X_{11}^{\,3} \times n_{11} + X_{12}^{\,3} \times n_{12}}{n_1 + n_2 + n_3 + ... + n_{11} + n_{12}}$

- $$M_S^4 = \frac{\sum_{i=1}^{n=12} X_i^4 \times n_i}{\sum_{i=1}^{12} n_i} = \frac{X_1^4 \times n_1 + X_2^4 \times n_2 + X_3^4 \times n_3 + \ldots\ldots\ldots + X_{11}^4 \times n_{11} + X_{12}^4 \times n_{12}}{n_1 + n_2 + n_3 + \ldots\ldots\ldots + n_{11} + n_{12}}$$

- $$M_C^1 = \frac{\sum_{i=1}^{n=12} \left(X_i - \overline{X}\right)^1 \times n_i}{\sum_{i=1}^{12} n_i} = \frac{\left(X_1 - \overline{X}\right)^1 \times n_1 + \left(X_2 - \overline{X}\right)^1 \times n_2 + \left(X_3 - \overline{X}\right)^1 \times n_3 + \ldots\ldots\ldots + \left(X_{12} - \overline{X}\right)^1 \times n_{12}}{n_1 + n_2 + n_3 + \ldots\ldots\ldots + n_{11} + n_{12}}$$

- $$M_C^2 = \frac{\sum_{i=1}^{n=12} \left(X_i - \overline{X}\right)^2 \times n_i}{\sum_{i=1}^{12} n_i} = \frac{\left(X_1 - \overline{X}\right)^2 \times n_1 + \left(X_2 - \overline{X}\right)^2 \times n_2 + \left(X_3 - \overline{X}\right)^2 \times n_3 + \ldots\ldots\ldots + \left(X_{12} - \overline{X}\right)^2 \times n_{12}}{n_1 + n_2 + n_3 + \ldots\ldots\ldots + n_{11} + n_{12}}$$

- $$M_C^3 = \frac{\sum_{i=1}^{n=12} \left(X_i - \overline{X}\right)^3 \times n_i}{\sum_{i=1}^{12} n_i} = \frac{\left(X_1 - \overline{X}\right)^3 \times n_1 + \left(X_2 - \overline{X}\right)^3 \times n_2 + \left(X_3 - \overline{X}\right)^3 \times n_3 + \ldots\ldots\ldots + \left(X_{12} - \overline{X}\right)^3 \times n_{12}}{n_1 + n_2 + n_3 + \ldots\ldots\ldots + n_{11} + n_{12}}$$

✍ التمـريـن 25.10:

لتكن القيم التالية:

34 12 45 68 100 76 54 90 91 66 23 18 17 14 18

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال القيم أعلاه في جدول وحيد البعد (شعاع) *Vect*.
2. تحديد أكبر قيمة (Le Maximum) من بين قيم الشعاع *Vect*.
3. نشر أكبر قيمة التي تم التوصل إليها في السؤال 2 كما يلي:

Le Maximum Egal :

✍ التمـريـن 26.10:

أكتب البرنامج الذي يسمح بـ:

1. إدخال n قيمة في جدول وحيد البعد (شعاع) T
2. حساب أكبر قيمة (Le Maximum) من بين قيم الشعاع T

3. نشر أكبر قيمة التي تم التوصل إليها في السؤال 2 كما يلي:

Le Maximum Egal :

✍ التمــريــن 27.10:

لتكن القيم المقدمة من التمرين 25.10 أعلاه

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال القيم أعلاه في شعاع *Vecteur*.

2. تحديد أقل قيمة (Le Minimum) من بين قيم الشعاع *Vect*.

3. نشر أكبر قيمة التي تم التوصل إليها في السؤال 2 كما يلي:

Le Minimum Egal :

✍ التمــريــن 28.10:

أكتب البرنامج الذي يسمح بـ:

1. إدخال *n* قيمة في جدول وحيد البعد (شعاع) *Tableau*

2. حساب أقل قيمة (Le Minimum) من بين قيم الشعاع *Tableau*

3. نشر أكبر قيمة التي تم التوصل إليها في السؤال 2 كما يلي:

Le Minimum Egal :

✍ التمــريــن 29.10:

لتكن القيم التالية:

12 36 67 89 80 87 100 23 15 200 345 123 456 124 124

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال القيم أعلاه في جدول وحيد البعد (شعاع) *Vecteur*.

2. حساب مجموع قيم الخانات ذات الرتبة الزوجية Cases Paires

3. نشر مجموع قيم الخانات ذات الرتبة الزوجية كما يلي:

La Somme Des Eléments Des Cases Paires Egale :

✍ التمــريــن 30.10:

أكتب البرنامج الذي يسمح بـ:

1. إدخال n قيمة في جدول وحيد البعد (شعاع) *Vecteur*.
2. حساب مجموع قيم الخانات ذات الرتبة الزوجية Cases Paires
3. نشر مجموع قيم الخانات ذات الرتبة الزوجية

✍ التمــريــن 31.10:

أكتب البرنامج الذي يسمح بـ:

1. إدخال التمرين 29.10 أعلاه في جدول وحيد البعد (شعاع) *Vecteur*.
2. حساب مجموع قيم الخانات ذات الرتبة الفردية Cases Impaires
3. نشر مجموع قيم الخانات ذات الرتبة الفردية كما يلي:

La Somme Des Eléments Des Cases Impaires Egale :

✍ التمــريــن 32.10:

أكتب البرنامج الذي يسمح بـ:

1. إدخال n قيمة في جدول وحيد البعد (شعاع) *Vecteur*.
2. حساب مجموع قيم الخانات ذات الرتبة الفردية Cases Paires.
3. نشر مجموع قيم الخانات ذات الرتبة الفردية.

✍ التمــريــن 33.10:

لتكن القيم التالية:

34 20 90 89 67 200 115 123 5 3 21 78 4 98 45

- **المطلــوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال القيم أعلاه في جدول وحيد البعد (شعاع) *Vecteur*.
2. ترتيب قيم الشعاع *Vecteur* ترتيبا تصاعديا من الأصغر إلى الأكبر.
3. نشر قيم الشعاع *Vecteur* مرتبة ترتيبا تصاعديا.

✍ التمــريــن 34.10:

أكتب البرنامج الذي يسمح بـ:

1. إدخال n قيمة في جدول وحيد البعد (شعاع) *Vecteur*.

2. ترتيب قيم الشعاع *Vecteur* ترتيبا تصاعديا من الأصغر إلى الأكبر.

3. نشر قيم الشعاع *Vecteur* مرتبة ترتيبا تصاعديا.

✍ التمـريـن 35.10:

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال قيم التمرين 33.10 أعلاه في جدول وحيد البعد (شعاع) *Vecteur*.

2. ترتيب قيم الشعاع *Vecteur* ترتيبا تنازليا من الأكبر إلى الأصغر.

3. نشر قيم الشعاع *Vecteur* مرتبة ترتيبا تنازليا.

✍ التمـريـن 36.10:

الجدول أدناه يقدم قيم حول المتغيرين X و Y

Y	16	18	23	24	28	29	26	31	32	34
X	20	24	28	22	32	28	32	36	41	41

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال قيم X في شعاع.

2. إدخال قيم Y في شعاع.

3. حساب المتوسط الحسابي $\overline{X}$ للمتغير الإحصائي X. حيث أن:

$$\overline{X} = \frac{\sum_{i=1}^{N=10} X_i}{N=10} = \frac{X_1 + X_2 + X_3 + \ldots\ldots\ldots + X_9 + X_{10}}{N=10}$$

4. حساب المتوسط الحسابي $\overline{Y}$ للمتغير الإحصائي Y. حيث أن:

$$\overline{Y} = \frac{\sum_{i=1}^{N=10} Y_i}{N=10} = \frac{Y_1 + Y_2 + Y_3 + \ldots\ldots\ldots + Y_9 + Y_{10}}{N=10}$$

5. حساب قيم الثوابت α و β التي تمثل معاملات معادلة خط الانحدار $Y_i = \alpha - \beta X_i$. حيث أن:

- $\beta = \dfrac{\sum_{i=1}^{N=10} X_i Y_i - N\overline{X}\,\overline{Y}}{\sum_{i=1}^{N=10} X_i^2 - N\overline{X}^2}$
- $\alpha = \overline{Y} - \beta\overline{X}$

✍ التمـريـن 37.10:

ليكن لدينا الشعاع سطر (Vecteur Ligne) A و الشعاع عمود (Vecteur Colonne) B حيث أن:

$$A = (. \ . \ . \ . \ . \ . \ . \ .)_n \quad , \quad B = \begin{pmatrix} . \\ . \\ . \\ . \\ . \\ . \\ . \\ . \end{pmatrix}_n$$

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال قيم الشعاع A.
2. إدخال قيم الشعاع B.
3. حساب قيمة جداء الشعاعين A و B أي $A \times B$.
4. نشر قيمة الجداء.

✍ التمـريـن 38.10:

أكتب البرنامج الذي يسمح بـ:

1. إدخال قيمة و لتكن *Valeur*.
2. إدخال قيم شعاع و ليكن *Vecteur*.
3. مقارنة جميع قيم الشعاع *Vecteur* بالقيمة *Valeur* حيث يتم استبدال كل قيمة صغيرة من القيمة *Valeur* بالقيمة 0.
4. نشر قيم الشعاع *Vecteur* بالقيم القديمة أي التي تم إدخالها
5. نشر قيم الشعاع *Vecteur* بالقيم الجديدة

الفصـــل 🕮 الحـادي عشـــر

الجـداول ثنـائيـة البعـد (المصفوفات)

Les Tableaux A Deux Dimensions (Les Matrices)

أهداف الفصل:

بعد انتهائك من دراسة و الإطلاع بعناية على محتويات هذا الفصل، فإنك تستطيع الإلمام بما يلي:

- ماذا نعني بالجداول ثنائية البعد (المصفوفات).
- بعد الجداول ثنائية البعد (المصفوفات).
- كيفية التصريح بالجداول ثنائية البعد (المصفوفات).
- مختلف الطرق المستخدمة في إدخال معطيات الجداول ثنائية البعد (المصفوفات).
- كيفية إدخال معطيات الجداول ثنائية البعد (المصفوفات).
- مختلق العمليات الممكن إنجازها على الجداول ثنائية البعد (المصفوفات).

01.11 – تمهيـد:

تم التطرق من خلال الفصل السابق إلى الجداول وحيد البعد الممثلة في الأشعة. هناك نوع آخر من الجداول و هي الجداول ثنائية البعد Tableau a Deux Dimensions و التي تشكل موضوع هذا الفصل.

02.11 – التعريف بالجداول ثنائيـة البعد (المصفوفـات):

المصفوفة عبارة عن جدول له عدد m من الأسطر و عدد n من الأعمدة يتضمن مجموعة من القيم أو العناصر.

1. المصفوفة A ذات $m=4$ سطر و $n=3$ عمود تأخذ الشكل التالي:

$$A=\begin{pmatrix}12 & 34 & 21\\ 1 & 23 & 78\\ 3 & 2 & 5\\ 10 & 0 & 45\end{pmatrix}_{4,3}$$

2. المصفوفة B ذات $m=2$ سطر و $n=5$ عمود تأخذ الشكل التالي:

$$B=\begin{pmatrix}142 & 36 & 1 & 0 & 100\\ 55 & 90 & 102 & 300 & 45\end{pmatrix}_{2,5}$$

3. المصفوفة M ذات $m=2$ سطر و $n=2$ عمود تأخذ الشكل التالي:

$$M=\begin{pmatrix}15 & 20\\ 23 & 67\end{pmatrix}_{2,2}$$

4. المصفوفة M_1 ذات $m=4$ سطر و $n=4$ عمود تأخذ الشكل التالي:

$$M_1=\begin{pmatrix}3 & 56 & 0 & 0\\ 45 & 6 & 1 & 8\\ 55 & 2 & 9 & 8\\ 12 & 22 & 11 & 56\end{pmatrix}_{4,4}$$

يتم تمثيل أي مصفوفة و لتكن mat ذات البعد $(5,7)$ أي ذات $m=5$ أسطر و $n=7$ أعمدة التالية:

$$mat=\begin{pmatrix}12 & 56 & 77 & 98 & 65 & 0 & 87\\ 23 & 55 & 43 & 12 & 65 & 78 & 90\\ 31 & 2 & 4 & 3 & 6 & 0 & 8\\ 14 & 45 & 11 & 10 & 20 & 30 & 80\\ 100 & 456 & 87 & 99 & 93 & 11 & 45\end{pmatrix}$$

على مستوى الذاكرة على شكل جدول ذو خمسة أسطر و سبعة أعمدة كما يلي:

12	56	77	98	65	0	87
23	55	43	12	65	78	90
31	2	4	3	6	0	8
14	45	11	10	20	30	80
100	456	87	99	93	11	45

انطلاقا من التمثيل أعلاه للمصفوفة *mat* يمكن اعتبار أن الجدول ثنائي البعد (المصفوفة) عبارة عن مجموعة من الخانات المتجاورة في الذاكرة، هذه الخانات تحمل نفس الاسم و الذي هو اسم المصفوفة الذي هو *mat*.

أي خانة من خانات المصفوفة *mat* و التي تحوي أو تتضمن قيمة معينة عبارة عن تقاطع سطر ذو رتبة معينة و عمود ذو رتبة معينة، حيث أن:

- الخانة التي تمثل تقاطع السطر رقم 1 و العمود رقم 1 يرمز لها بالرمز *mat*[1,1] و هي تحوي أو تتضمن القيمة 12 أي $mat[1,1]=12$
- الخانة التي تمثل تقاطع السطر رقم 3 و العمود رقم 5 يرمز لها بالرمز *mat*[3,5] و هي تحوي أو تتضمن القيمة 6 أي $mat[3,5]=6$

في المصفوفات نحتاج إلى مؤشرين i و j حيث أن:

- المؤشر i يتعلق أو خاص بالأسطر فهو يتغير من 1 إلى 5 أي $i=1,2,3,4,5$
- المؤشر j يتعلق أو خاص بالأعمدة فهو يتغير من 1 إلى 7 أي $i=1,2,3,4,5,6,7$

يتم الدخول إلى أي عنصر من عناصر المصفوفة باستخدام المؤشرين i و j كما يلي:

mat[*i*, *j*]: تمثل القيمة أو العنصر الذي يقع في الخانة التي تمثل تقاطع السطر رقم i و العمود رقم j

حيث أن:

- *mat*[1,3]: عبارة عن الخانة التي تمثل تقاطع السطر رقم 1 و العمود رقم 3 حيث أن: $mat[1,3]=77$
- *mat*[3,1]: عبارة عن الخانة التي تمثل تقاطع السطر رقم 3 و العمود رقم 1 حيث أن: $mat[3,1]=31$
- *mat*[4,6]: عبارة عن الخانة التي تمثل تقاطع السطر رقم 4 و العمود رقم 6 حيث أن: $mat[4,6]=30$

03.11 – بعد الجداول ثنائيـة البعد (المصفوفـات):

الذي يحدد بعد المصفوفة هو عدد الأسطر و عدد الأعمدة التي تتكون منها المصفوفة، و بالتالي نسمي بعد (Dimension) المصفوفة عدد الأسطر و عدد الأعمدة التي تحتويها أو تتضمنها المصفوفة.

مثـال 01.11:

- المصفوفة X تحتوي على $m=3$ أسطر و $n=4$ أعمدة و بالتالي فإن بعدها يساوي $X(3,4)$

$$X_{(3,4)}=\begin{pmatrix}4 & 3 & 6 & 0\\ 11 & 10 & 20 & 30\\ 87 & 99 & 93 & 11\end{pmatrix}_{(3,4)}$$

و تمثل كما يلي:

4	3	6	0
11	10	20	30
87	99	93	11

- المصفوفة Y تحتوي على $m=4$ أسطر و $n=2$ أعمدة و بالتالي فإن بعدها يساوي $Y(4,2)$

$$Y_{(4,2)}=\begin{pmatrix}98 & 65\\ 12 & 65\\ 3 & 6\\ 10 & 20\end{pmatrix}_{(4,2)}$$

و تمثل كما يلي:

98	65
12	65
3	6
10	20

- المصفوفة Z تحتوي على $m=7$ أسطر و $n=3$ أعمدة و بالتالي فإن بعدها يساوي $Z(7,3)$

$$Z_{(7,3)} = \begin{pmatrix} 98 & 65 & 0 \\ 12 & 65 & 78 \\ 3 & 6 & 0 \\ 10 & 20 & 30 \\ 99 & 93 & 11 \\ 88 & 79 & 70 \\ 44 & 10 & 50 \end{pmatrix}_{(7,3)}$$

و تمثل كما يلي:

98	65	0
12	65	78
3	6	0
10	20	30
99	93	11
88	79	70
44	10	50

04.11 – التصريح بالجداول ثنائيـة البعد (المصفوفـات):

مثلما يتم التصريح بالجداول وحيدة البعد (الأشعة)، يتم كذالك التصريح بالجداول ثنائية البعد (المصفوفات) حتى نتمكن من استخدامها في البرنامج. عملية التصريح هذه تتم بواسطة Le mot réservé **ARRAY** وفقا لطريقة الكتابة التالية:

Var** Nom_Du_Tableau **: array [1.. m , 1.. n] of** Type_ Du_Tableau **;

حيث أن:

- ***Var*** : يمثل Le Mot Reservé الذي يسمح بالتصريح باسم المصفوفة (الجدول ثنائي البعد).
- Nom_Du_Tableau : اسم الجدول ثنائي البعد (المصفوفة).
- **array** : يمثل Le Mot Reservé الذي يسمح بالتصريح بالمصفوفة.
- **n** : تمثل عدد أسطر المصفوفة.
- **n** : تمثل عدد أعمدة المصفوفة.
- Type_ Du_Tableau : يمثل نوع المصفوفة الذي نريد التصريح بها، نوع المصفوفة يتحدد بنوع المعطيات التي يحتويها، حيث أن Type_ Du_Tableau تأخذ أحد Les Identificateurs Prédéfinis التالية:

1. ***Integer*** : و التي تعني أن المصفوفة المصرح بها من النوع الصحيح، أي أن المعطيات (القيم) التي تحتويها المصفوفة من النوع الصحيح.
2. ***Real*** : و التي تعني أن المصفوفة المصرح بها من النوع الحقيقي، أي أن المعطيات (القيم) التي تحتويها المصفوفة من النوع الحقيقي.
3. ***String*** : و التي تعني أن المصفوفة المصرح بها من النوع سلسلة أحرف، أي أن المعطيات (القيم) التي تحتويها المصفوفة من النوع سلسلة أحرف.
4. ***Char*** : و التي تعني أن المصفوفة المصرح بها من النوع حرف، أي أن المعطيات (القيم) التي تحتويها المصفوفة من النوع حرف.

- **;** : فاصلة منقوطة Point Virgule

بغرض التوضيح كيفية التصريح بالمصفوفات نأخذ المثال التالي:

مثـال 02.11:

1. يتم التصريح بالمصفوفة من النوع الصحيح و التي تحمل الاسم *matr* ذات البعد (5,8) كما يلي:

Var** matr **: array [1..5,1..8] of** integer **;

حيث يتم على مستوى الذاكرة تخصيص جدول من 5 أسطر و 8 أعمدة و بالتالي 40 خانة فارغة

2. يتم التصريح بالمصفوفة من النوع الحقيقي و التي تحمل الاسم *f* ذات البعد (4,2) كما يلي:

Var** f **: array [1..4,1..2] of** real **;

حيث يتم على مستوى الذاكرة تخصيص جدول من 4 أسطر و 2 أعمدة و بالتالي 8 خانات فارغة

3. يتم التصريح بالمصفوفة من النوع سلسلة أحرف و التي تحمل الاسم *t* ذات البعد (20,2) كما يلي:

Var** t **: array [1..20,1..2] of** string **;

حيث يتم على مستوى الذاكرة تخصيص جدول من 20 سطر و 2 أعمدة و بالتالي 40 خانة فارغة

4. يتم التصريح بالمصفوفة من النوع حرف و التي تحمل الاسم *c* ذات البعد (5,4) كما يلي:

Var** c **: array [1..5,1..4] of** char **;

حيث يتم على مستوى الذاكرة تخصيص جدول من 5 أسطر و 4 أعمدة و بالتالي 20 خانة فارغة

05.11 – إدخـال المعطيـات فـي الجداول ثنـائيـة البعد (المصفـوفـات):

بعد مرحلة أو عملية التصريح بالجدول ثنائي البعد (المصفوفة)، ننتقل إلى عملية أو مرحلة إدخال قيم و عناصر (معطيات) هذه المصفوفة. عملية إدخال معطيات المصفوفة يمكن لها أن تتم بطريقتين[1] هما:

1. إدخال معطيات المصفوفة عن طريق البرنامج.
2. إدخال معطيات المصفوفة عن لوحة المفاتيح.

سوف يتم التطرق بالتفصيل إلى كل طريقة أدناه:

01.05.11 – إدخـال المعطيـات عن طريـق البرنامـج:

تبعا لهذه الطريقة لإدخال قيم و عناصر (معطيات) المصفوفة، فإن هذه المعطيات (مهما كان نوع المتغيرة) تكون موجودة في أو داخل البرنامج أي يتم توجهيها مباشرة في خانات المصفوفة الموافقة لها عبر البرنامج مثلما توضحه الأمثلة التالية:

مثـال 03.11:

لتكن المصفوفة من النوع الصحيح mat_1 ذات البعد $(2,3)$ و التي تساوي ما يلي:

$$mat_1 = \begin{pmatrix} 13 & 45 & 7 & 9 \\ 34 & 20 & 10 & 11 \\ 71 & 40 & 78 & 96 \end{pmatrix}_{(3,4)}$$

يتم إدخال معطيات الشعاع mat_1 عن طريق البرنامج كما يلي:

```
mat_1 [1,1]:=13;
mat_1 [1,2]:=45;
mat_1 [1,3]:=7;
mat_1 [1,4]:=13;
mat_1 [2,1]:=34;
mat_1 [2,2]:=20;
mat_1 [2,3]:=10;
mat_1 [2,4]:=11;
mat_1 [3,1]:=71;
```

[1] لمزيد من المعلومات حول طرق إدخال المعطيات أنظر الفصل الثالث

```
mat_1 [3,2]:=40;
mat_1 [3,3]:=78;
mat_1 [3,4]:=96;
```

و عليه يصبح البرنامج الذي يسمح بالتصريح بالشعاع *mat_1* و إدخال معطياته هو التالي:

```
program MATRICE_ENTIERE ;
var mat_1:array[1..3,1..4] of integer;
begin
   mat_1 [1,1]:=13;
   mat_1 [1,2]:=45;
   mat_1 [1,3]:=7;
   mat_1 [1,4]:=13;
   mat_1 [2,1]:=34;
   mat_1 [2,2]:=20;
   mat_1 [2,3]:=10;
   mat_1 [2,4]:=11;
   mat_1 [3,1]:=71;
   mat_1 [3,2]:=40;
   mat_1 [3,3]:=78;
   mat_1 [3,4]:=96;
end.
```

يقوم الحاسوب بتنفيذ البرنامج أعلاه كما يلي:

1. عند تنفيذ السطر *var mat_1:array[1..3,1..4] of integer;* يقوم الحاسوب على مستوى الذاكرة بتخصيص جدول يتكون من 3 أسطر و 4 أعمدة أي 12 خانة فارغة كما يلي:

mat_1

2. عند تنفيذ السطر ***mat_1 [1,1]:=13;*** يقوم بتخزين القيمة الصحيحة 13 في الخانة *mat_1*[1,1] كما هو موضح أدناه:

	13			
mat _1				

3. عند تنفيذ السطر ***mat_1 [1,2]:=45;*** يقوم بتخزين القيمة الصحيحة 45 في الخانة $mat_1[1,2]$ كما هو موضح أدناه:

	13	45		
mat _1				

4. و هكذا حتى يتم آخر سطر لإدخال معطيات المصفوفة ***mat_1 [3,4]:=96;*** فيتم الحصول على مستوى الذاكرة على ما يلي:

	13	45	7	9
mat _1	34	20	10	11
	71	40	78	96

مثـال 04.11:

لتكن المصفوفة من النوع الحقيقي mat_2 ذات البعد (3,3) و التي تساوي ما يلي:

$$mat_2 = \begin{pmatrix} 2.2 & 0.33 & 67.87 \\ 0.056 & 7.8 & 0.98 \\ 1.2 & 3.1 & 76.098 \end{pmatrix}_{(3,3)}$$

يتم إدخال معطيات الشعاع mat_2 عن طريق البرنامج كما يلي:

```
mat_2 [1,1]:=2.2;
mat_2 [1,2]:=0.33;
mat_2 [1,3]:=67.87;
mat_2 [2,1]:=0.056;
mat_2 [2,2]:=7.8;
mat_2 [2,3]:=0.98;
mat_2 [3,1]:=1.2;
mat_2 [3,2]:=3.1;
```

mat_2 [3,3]:=76.098;

و عليه يصبح البرنامج الذي يسمح بالتصريح بالشعاع *mat_2* و إدخال معطياته هو التالي:

```
program MATRICE_REELLE ;
var mat_2:array[1..3,1..3] of real;
begin
   mat_2 [1,1]:=2.2;
   mat_2 [1,2]:=0.33;
   mat_2 [1,3]:=67.87;
   mat_2 [2,1]:=0.056;
   mat_2 [2,2]:=7.8;
   mat_2 [2,3]:=0.98;
   mat_2 [3,1]:=1.2;
   mat_2 [3,2]:=3.1;
   mat_2 [3,3]:=76.098;
end.
```

يقوم الحاسوب بتنفيذ البرنامج أعلاه كما يلي:

1. عند تنفيذ السطر *var mat_2:array[1..3,1..3] of real;* يقوم الحاسوب على مستوى الذاكرة بتخصيص جدول يتكون من 3 أسطر و 3 أعمدة أي 9 خانة فارغة كما يلي:

mat_2

2. عند تنفيذ السطر ***mat_2 [1,1]:=2.2;*** يقوم بتخزين القيمة الصحيحة 2.2 في الخانة *mat_1*[1,1] كما هو موضح أدناه:

mat_2

2.2		

3. عند تنفيذ السطر ***mat_2 [1,2]:=0.33;*** يقوم بتخزين القيمة الصحيحة 0.33 في الخانة $mat_1[1,2]$ كما هو موضح أدناه:

mat_2

2.2	0.33	

4. و هكذا حتى يتم آخر سطر لإدخال معطيات المصفوفة ***mat_2 [3,3]:=76.098;*** فيتم الحصول على مستوى الذاكرة على ما يلي:

mat_2

2.2	0.33	67.87
0.056	7.8	0.98
1.2	3.1	76.098

مثـال 05.11:

لتكن المصفوفة من النوع سلسلة أحرف *nom* ذات البعد $(6,2)$ و التي تساوي ما يلي:

$$nom = \begin{pmatrix} ZAHRA & YA \\ SIHEM & MENAD \\ IMENE & YAMINI \\ FAIZA & MAAZOUZI \\ DJASMINE & KHALIL \\ KHALDIA & BENSALEM \end{pmatrix}$$

البرنامج الذي يسمح بإدخال معطيات المصفوفة *nom* هو التالي:

```
program MATRICE_CHAINE_CARACTERES ;
var nom : array[1..6,1..2] of string;
begin
   nom [1,1]:='ZAHRA';
   nom [1,2]:='YA';
   nom [2,1]:= 'SIHEM';
   nom [2,2]:= 'MENAD';
   nom [3,1]:= 'IMENE';
   nom [3,2]:= 'YAMINI';
```

```
nom [4,1]:= 'FAIZA';
nom [4,2]:= 'MAZOUZI';
nom [5,1]:= 'DJASMINE';
nom [5,2]:= 'KHALIL';
nom [6,1]:= 'KHALDIA';
nom [6,2]:= 'BENSALEM';
end.
```

يقوم الحاسوب بتنفيذ البرنامج أعلاه كما يلي:

5. عند تنفيذ السطر *var nom : array[1..6,1..2] of string;* يقوم الحاسوب على مستوى الذاكرة بتخصيص جدول يتكون من 6 أسطر و 2 أعمدة أي 12 خانة فارغة كما يلي:

nom

1. عند تنفيذ السطر ***nom [1,1]:='ZAHRA';*** يقوم بتخزين المعطية من النوع سلسلة أحرف *ZAHRA* في الخانة [1,1]*nom* كما هو موضح أدناه:

ZAHRA	

2. عند تنفيذ السطر ***nom [1,2]:='YA';*** يقوم بتخزين المعطية من النوع سلسلة أحرف *SIHEM* في الخانة [1,2]*nom* كما هو موضح أدناه:

ZAHRA	*YA*

3. و هكذا حتى يتم تنفيذ آخر تعليمة إدخال معطيات المصفوفة ***nom [6,2]:= 'BENSALEM';*** فيتم الحصول على مستوى الذاكرة على ما يلي:

ZAHRA	*YA*
SIHEM	*MENAD*
IMENE	*YAMINI*
FAIZA	*MAZOUZI*
DJASMINE	*KHALIL*
KHALDIA	*BENSALEM*

مثـال 06.11:

لتكن المصفوفة من النوع حرف *ch* ذات البعد $(6,3)$ و التي تساوي ما يلي:

$$ch = \begin{pmatrix} Z \\ S \\ I \\ F \\ D \end{pmatrix}$$

البرنامج الذي يسمح بإدخال معطيات المصفوفة *ch* هو التالي:

```
program MATRICE_CARACTERE ;
var ch : array[1..6,1..3] of char;
begin
  ch [1,1]:='Z'; ch [1,2]:='Y'; ch [1,3]:='S';
  ch [2,1]:='S'; ch [2,2]:='M'; ch [2,3]:='T';
  ch [3,1]:='I'; ch [3,2]:='Y'; ch [3,3]:='T';
  ch [4,1]:='F'; ch [4,2]:='M'; ch [4,3]:='C';
```

```
ch [5,1]:='D'; ch [5,2]:='K'; ch [5,3]:='F';
ch [6,1]:='K'; ch [6,2]:='B'; ch [6,3]:='R';
end.
```

يقوم الحاسوب بتنفيذ البرنامج أعلاه كما يلي:

1. عند تنفيذ السطر *var ch : array[1..6,1..3] of char;* يقوم الحاسوب على مستوى الذاكرة بتخصيص جدول مكون من 6 أسطر و 3 أعمدة أي 18 خانة فارغة كما يلي:

2. عند تنفيذ السطر ***ch [1,1]:='Z';*** يقوم بتخزين المعطية من النوع حرف *Z* في الخانة [1,1]*ch* كما هو موضح أدناه:

Z		

3. عند تنفيذ السطر ***ch [1,2]:='Y';*** يقوم بتخزين المعطية من النوع حرف *Y* في الخانة [1,2]*ch* كما هو موضح أدناه:

Z	*Y*	

4. و هكذا حتى يتم تنفيذ جميع أسطر إدخال معطيات المصفوفة فيتم الحصول على مستوى الذاكرة على ما يلي:

Z	*Y*	*S*
S	*M*	*T*
I	*Y*	*T*
F	*M*	*C*
D	*K*	*F*
K	*B*	*R*

02.05.11 – إدخـال المعطيـات عن طريـق لوحـة المفاتيـح:

إضافة إلى طريقة إدخال معطيات المصفوفة عن طريق البرنامج و التي مضمونها أن المعطيات تكون موجودة في البرنامج أي يتم توجهيها مباشرة في الخانات الموافقة لها عبر البرنامج، هناك طريقة أخرى لإدخال المعطيات و هي الأكثر استخداما و التي تتمثل في إدخال معطيات المصفوفة عن طريق لوحة المفاتيح (Par Le Clavier). مضمون هذه الطريقة هو أنه يتم إدخال معطيات المصفوفة بعد تنفيذ البرنامج حيث يطلب الحاسوب من المستخدم إدخال هذه المعطيات ليقوم هذا الأخير بإدخالها عن طريق لوحة المفاتيح. يتم إدخال معطيات المصفوفة عن طريق لوحة المفاتيح باستخدام ***Read*** أو ***Readln*** كما توضحه الأمثلة التالية:

مثـال 07.11:

لتكن المصفوفة من النوع الصحيح m ذات البعد (3,4) و التي تساوي ما يلي:

$$m = \begin{pmatrix} 125 & 43 & 12 & 31 \\ 90 & 45 & 11 & 21 \\ 34 & 32 & 61 & 50 \end{pmatrix}_{(3,4)}$$

يتم إدخال معطيات المصفوفة m عن طريق لوحة المفاتيح باستخدام ***Read*** أو ***Readln*** كما يلي:

```
Readln ( m[1,1] ) ;
Readln ( m[1,2] ) ;
Readln ( m[1,3] ) ;
Readln ( m[1,4] ) ;
Readln ( m[2,1] ) ;
Readln ( m[2,2] ) ;
```

```
Readln ( m[2,3] ) ;
Readln ( m[2,4] ) ;
Readln ( m[3,1] ) ;
Readln ( m[3,2] ) ;
Readln ( m[3,3] ) ;
Readln ( m[3,4] ) ;
```

و عليه يصبح البرنامج الذي يسمح بالتصريح بالمصفوفة m و إدخال معطياته هو التالي:

```
program Inroduction_Des_Elements_MATRICE_Par_CLAVIER;
var m:array[1..3,1..4] of integer;
begin
    Readln ( m[1,1] ) ;
    Readln ( m[1,2] ) ;
    Readln ( m[1,3] ) ;
    Readln ( m[1,4] ) ;
    Readln ( m[2,1] ) ;
    Readln ( m[2,2] ) ;
    Readln ( m[2,3] ) ;
    Readln ( m[2,4] ) ;
    Readln ( m[3,1] ) ;
    Readln ( m[3,2] ) ;
    Readln ( m[3,3] ) ;
    Readln ( m[3,4] ) ;
end.
```

ما يلاحظ أو يقال عن البرنامج الذي يسمح بإدخال عناصر المصفوفة أنه يحتوي على العديد من الأسطر و بغرض تقليص عدد أسطر البرنامج نقوم باستخدام إحدى الحلقات التي تم التطرق إليها في الفصل السابق سوف نقتصر على استخدام الحلقة ... *For ... to ... do* كما يلي:

المصفوفة m ذات البعد (3,4) لها مؤشرين هما:

- المؤشر i **خاص بالأسطر** يتغير من 1 إلى 3 أي $i = 1,2,3$ و عليه يتم استخدام الحلقة كما يلي:

```
For i:=1 to 3 do
```

- المؤشر j **خاص بالأعمدة** يتغير من 1 إلى 4 أي $j = 1,2,3,4$ و عليه يتم استخدام الحلقة كما يلي:

For j:=1 to 4 do

يتم إدخال عناصر المصفوفة m باستخدام الحلقة *For ... to ... do ...* كما يلي:

```
program Inroduction_Des_Elements_MATRICE_Par_CLAVIER;
var m:array[1..3,1..4] of integer;
var i ,j: integer ;
begin
    for i:= 1 to 3  do
    begin
        for j:= 1 to 4 do
        Readln ( m[ i , j ] ) ;
    end;
end.
```

كما يمكن إعادة كتابة البرنامج على الشكل التالي:

```
program Inroduction_Des_Elements_MATRICE_Par_CLAVIER;
var m:array[1..3,1..4] of integer;
var i ,j: integer ;
begin
{INTRODUCTION DES ELEMENTS DE LA MATRICE  A } ;
for i:= 1 to 3  do
begin
   for j:= 1 to 4 do
   begin
      write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  M :') ;
      Readln ( m[ i , j ] ) ;
   end;
end;
{AFFIGAGE DES ELEMENTS DE LA MATRICE  A } ;
for i:= 1 to 3 do
begin
```

```
for j:=1 to 4 do
write ( m[i,j],' ' ) ;
writeln;
end;
readln ;
end.
```

عند تنفيذ البرنامج نحصل على الشاشة على ما يلي:

```
Donnez l élément Numéro [1,1] De La Matrice  A : 125
Donnez l élément Numéro [1,2] De La Matrice  A : 43
Donnez l élément Numéro [1,3] De La Matrice  A : 12
Donnez l élément Numéro [1,4] De La Matrice  A : 31
Donnez l élément Numéro [2,1] De La Matrice  A : 90
Donnez l élément Numéro [2,2] De La Matrice  A : 45
Donnez l élément Numéro [2,3] De La Matrice  A : 11
Donnez l élément Numéro [2,4] De La Matrice  A : 21
Donnez l élément Numéro [3,1] De La Matrice  A : 34
Donnez l élément Numéro [3,2] De La Matrice  A : 32
Donnez l élément Numéro [3,3] De La Matrice  A : 61
Donnez l élément Numéro [3,4] De La Matrice  A : 50
```

مثـال 08.11:

لتكن المصفوفة من النوع سلسلة أحرف *ville* ذات البعد (5,2) و التي تتضمن مجموعة من المدن و الجهة التي تقع فيها

$$ville = \begin{pmatrix} TIARET & OUEST \\ SETIF & EST \\ SKIKDA & EST \\ SAIDA & OUEST \\ ORAN & OUEST \end{pmatrix}$$

يتم إدخال معطيات المصفوفة *ville* عن طريق لوحة المفاتيح باستخدام *Read* أو *Readln* كما يلي:

```
program Inroduction_Des_Elements_MATRICE_Par_CLAVIER;
var ville :array[1..5,1..2] of string;
var i ,j: integer ;
```

```
begin
{INTRODUCTION DES ELEMENTS DE LA MATRICE  VILLE } ;
for i:= 1 to 5  do
begin
   for j:= 1 to 2 do
   begin
     write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  VILLE :') ;
      Readln ( ville[ i , j ] ) ;
   end;
end;
{AFFICHAGE DES ELEMENTS DE LA MATRICE  VILLE } ;
for i:= 1 to 3 do
 begin
   for j:=1 to 4 do
   write ( ville[i,j],'  ' ) ;
   writeln;
 end;
readln ;
end.
```

06.11 – العمليـات علـى الجداول ثنـائيـة البعد (المصفوفـات):

هناك العديد من العمليات الممكن إجراءها على المصفوفات من نفس البعد نذكر على سبيل المثال لا الحصر العمليات التالية:

1. جمع مصفوفتين.
2. فرق مصفوفتين.
3. جداء مصفوفتين.
4. حساب مجموع عناصر مصفوفة معينة.

من خلال هذه الفقرة سوف يتم التطرق إلى بعض هذه العمليات

01.06.11 – جمـع مصفوفتيـن:

لتكن المصفوفتين A و B من النوع الصحيح و لهما نفس البعد $(4,3)$ حيث أن:

$$A=\begin{pmatrix}22 & 15 & 9\\ 11 & 77 & 45\\ 43 & 23 & 55\\ 37 & 0 & 0\end{pmatrix}_{(4,3)} \quad , \quad B=\begin{pmatrix}122 & 215 & 39\\ 411 & 577 & 645\\ 743 & 823 & 955\\ 337 & 110 & 100\end{pmatrix}_{(4,3)}$$

جمع المصفوفتين A و B عبارة عن مصفوفة ثالثة و ليكن C لها نفس بعدهما و الذي يساوي $(4,3)$ حيث أن:

$$C=A+B$$

بغرض كتابة البرنامج الذي يسمح بجمع المصفوفتين A و B في المصفوفة C نتبع الخطوات التالية:

1. التصريح بالمصفوفة A ذات البعد $(4,3)$ من النوع الصحيح: يتم ذالك كما يلي

```
var a : array[1..4,1..3] of integer ;
```

2. التصريح بالمصفوفة B ذات البعد $(4,3)$ من النوع الصحيح: يتم ذالك كما يلي

```
var b : array[1..4,1..3] of integer ;
```

3. التصريح بالمصفوفة C ذات البعد $(4,3)$ من النوع الصحيح: يتم فيه تخزين نتيجة جمع عناصر المصفوفتين A و B

```
var c : array[1..4,1..3] of integer ;
```

4. التصريح بالمتغيرتين i و i من النوع الصحيح و التي تستخدم كمؤشرات للمصفوفات الثلاثة

```
var i,j : integer ;
```

5. إدخال معطيات أو عناصر المصفوفة A: يتم ذالك كما يلي

```
{INTRODUCTION DES ELEMENTS DE LA MATRICE  A } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
   begin
    write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  A :') ;
     Readln ( a[ i , j ] ) ;
   end;
end;
```

6. إدخال معطيات أو عناصر الشعاع B: يتم ذالك كما يلي

```
{INTRODUCTION DES ELEMENTS DE LA MATRICE  B } ;
```

```
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
   begin
    write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  B :') ;
     Readln ( b[ i , j ] ) ;
   end;
end;
```

7. حساب قيم المصفوفة C كما يلي:

```
{CALCUL DES ELEMENTS DE LA MATRICE  C } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
     c[ i , j ] := a[ i , j ] +  b[ i , j ] ) ;
end;
```

8. نشر عناصر الشعاع C كما يلي:

```
{AFFICHAGE DES ELEMENTS DE LA MATRICE  C } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
       write(c[ i , j ] ,'  ') ;
  writeln ;
end;
```

بتجميع ما تم التوصل إليه أعلاه نحصل على البرنامج التالي:

```
program SOMME_De_DEUX_MATRICES;
var a :array[1..4,1..3] of integer;
   b :array[1..4,1..3] of integer;
   c :array[1..4,1..3] of integer;
var i ,j: integer ;
begin
{INTRODUCTION DES ELEMENTS DE LA MATRICE  A } ;
for i:= 1 to 4  do
begin
```

```
   for j:= 1 to 3 do
   begin
    write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  A :') ;
     Readln ( a[ i , j ] ) ;
   end;
end;
{INTRODUCTION DES ELEMENTS DE LA MATRICE  B } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
   begin
    write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  B :') ;
     Readln ( b[ i , j ] ) ;
   end;
end;
{CALCUL DES ELEMENTS DE LA MATRICE  C } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
     c[ i , j ] := a[ i , j ] +  b[ i , j ]  ;
end;
{AFFICHAGE DES ELEMENTS DE LA MATRICE  C } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
       write(c[ i , j ] ,'  ') ;
  writeln ;
end;
readln ;
end.
```

يمكن التقليص من عدد أسطر البرنامج عن طريق استخدام حلقة واحدة كما يلي:

```
program SOMME_De_DEUX_MATRICES;
var a :array[1..4,1..3] of integer;
   b :array[1..4,1..3] of integer;
   c :array[1..4,1..3] of integer;
var i ,j: integer ;
begin
{INTRODUCTION DE  A et B  et  CALCUL DE  C } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
   begin
```

```
      {INTRODUCTION DES ELEMENTS DE LA MATRICE  A } ;
      write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  A :') ;
       Readln ( a[ i , j ] ) ;
     {INTRODUCTION DES ELEMENTS DE LA MATRICE  B } ;
      write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  B :') ;
       Readln ( b[ i , j ] ) ;
     {CALCUL DES ELEMENTS DE LA MATRICE  C } ;
       c[ i , j ] := a[ i , j ] +  b[ i , j ]  ;
    end;
end;
{AFFICHAGE DES ELEMENTS DE LA MATRICE  C } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
        write(c[ i , j ] ,'  ') ;
  writeln ;
end;
readln ;
end.
```

02.06.11 – فـرق مصفوفتيـن:

البرنامج الذي يسمح بحساب فرق مصفوفتين A و B هو نفسه الذي يسمح بحساب مجموعهما الذي يتغير هو السطر التالي

```
c[ i , j ] := a[ i , j ] +  b[ i , j ]  ;
```

و الذي يتم استبداله بالسطر التالي:

```
c[ i , j ] := a[ i , j ] -  b[ i , j ]  ;
```

و عليه يصبح البرنامج هو التالي:

```
program SOMME_De_DEUX_MATRICES;
var a :array[1..4,1..3] of integer;
   b :array[1..4,1..3] of integer;
   c :array[1..4,1..3] of integer;
var i ,j: integer ;
begin
{INTRODUCTION DE  A et B  et  CALCUL DE  C } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
   begin
```

```
        {INTRODUCTION DES ELEMENTS DE LA MATRICE  A } ;
        write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  A :') ;
        Readln ( a[ i , j ] ) ;
       {INTRODUCTION DES ELEMENTS DE LA MATRICE  B } ;
        write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  B :') ;
        Readln ( b[ i , j ] ) ;
       {CALCUL DES ELEMENTS DE LA MATRICE  C } ;
        c[ i , j ] := a[ i , j ] - b[ i , j ]  ;
      end;
    end;
    {AFFICHAGE DES ELEMENTS DE LA MATRICE  C } ;
    for i:= 1 to 4  do
    begin
      for j:= 1 to 3 do
          write(c[ i , j ] ,'  ') ;
     writeln ;
    end;
    readln ;
    end.
```

03.06.11 – جداء مصفوفتيـن:

حتى يتم إنجاز عملية جداء مصفوفتين يشترط على المصفوفة الأولى أن يكون عدد أعمدتها مساوي لعدد أسطر المصفوفة الثانية، نتيجة عملية جداء مصفوفتين عبارة عن مصفوفة ثالثة عدد أسطرها يساوي عدد أسطر المصفوفة الأولى و عدد أعمدتها يساوي عدد أعمدة المصفوفة الثانية. و عليه فإنه حتى نتمكن من إجراء جداء المصفوفتين $B_{(n,k)}$ و $A_{(m,n)}$ و الذي يرمز له بالرمز $A \times B$ يشترط أن:

- عدد أعمدة المصفوفة الأولى $A_{(m,n)}$ الذي هو n **يساوي** عدد أسطر المصفوفة الثانية $B_{(n,k)}$ الذي هو n

نتيجة جداء المصفوفتين $A_{(m,n)}$ و $B_{(n,k)}$ عبارة عن مصفوفة ثالثة و لتكن C عدد أسطرها يساوي عدد أسطر المصفوفة الأولى و عدد أعمدتها يساوي عدد أعمدة المصفوفة الثانية أي $C_{(m,k)}$.

عملية الجداء تتم كما يوضحه المثال التالي:

مثـال 09.11:

لتكن المصفوفتين B و A التاليتين:

$$A_{(3,4)} = \begin{pmatrix} 2 & 5 & 1 & 3 \\ 4 & 6 & 8 & 10 \\ 1 & 4 & 2 & 8 \end{pmatrix} \quad , \quad B_{(4,5)} = \begin{pmatrix} 10 & 20 & 30 & 40 & 50 \\ 0 & 8 & 6 & 4 & 2 \\ 1 & 4 & 5 & 10 & 12 \\ 2 & 3 & 15 & 20 & 2 \end{pmatrix}$$

$A \times B$ عبارة عن مصفوفة C أي:

$$A_{(3,4)} \times B_{(4,5)} = C_{(3,5)}$$

حيث أن المصفوفة C تساوي ما يلي:

$$C = \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} \end{pmatrix}$$

حيث أن:

- $C_{11} = 2 \times 10 + 5 \times 0 + 1 \times 1 + 3 \times 2 = 27$
- $C_{12} = 2 \times 20 + 5 \times 8 + 1 \times 4 + 3 \times 3 = 93$
- $C_{13} = 2 \times 30 + 5 \times 6 + 1 \times 5 + 3 \times 15 = 140$
- $C_{13} = 2 \times 40 + 5 \times 4 + 1 \times 10 + 3 \times 20 = 170$
- $C_{15} = 2 \times 50 + 5 \times 2 + 1 \times 12 + 3 \times 2 = 128$

بغرض كتابة البرنامج الذي يسمح بحساب $A \times B$ نتبع الخطوات التالية:

1. التصريح بالمصفوفة A من النوع الصحيح ذات البعد $A_{(3,4)}$
2. التصريح بالمصفوفة B من النوع الصحيح ذات البعد $B_{(4,5)}$
3. التصريح بالمصفوفة C من النوع الصحيح ذات البعد $C_{(3,5)}$
4. التصريح بالمتغيرات i و j و k من النوع الصحيح و التي تستخدم كمؤشرات للمصفوفات
5. إدخال عناصر المصفوفة A :
6. إدخال عناصر المصفوفة B :
7. حساب عناصر المصفوفة C كما يلي:

```
for i:= 1 to 3  do
begin
   for j:= 1 to 5 do
   begin
      c[ i , j ] := 0 ;
      for k:= 1 to 4 do
```

```
            c[ i , j ] := c[ i , j ] + a[ i , k ] * b[ k , j ]  ;
        end;
    end;
```

8. نشر المصفوفة C :

و عليه يصبح البرنامج هو التالي:

```
program SOMME_De_DEUX_MATRICES;
var a :array[1..3,1..4] of integer;
   b :array[1..4,1..5] of integer;
   c :array[1..3,1..5] of integer;
var i ,j , k : integer ;
begin
{INTRODUCTION DES ELEMENTS DE LA MATRICE  A } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 3 do
   begin
     write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  A :') ;
      Readln ( a[ i , j ] ) ;
   end;
end;
{INTRODUCTION DES ELEMENTS DE LA MATRICE  B } ;
for i:= 1 to 4  do
begin
   for j:= 1 to 5 do
   begin
     write('Donnez l élément Numéro [', i ,',',j,'] De La Matrice  B :') ;
      Readln ( b[ i , j ] ) ;
   end;
end;
{CALCUL DES ELEMENTS DE LA MATRICE  C } ;
for i:= 1 to 3  do
begin
   for j:= 1 to 5 do
   begin
      c[ i , j ] := 0 ;
      for k:= 1 to 4 do
       c[ i , j ] := c[ i , j ] + a[ i , k ] * b[ k , j ]  ;
```

```
  end;
end;
{AFFICHAGE DES ELEMENTS DE LA MATRICE  C } ;
for i:= 1 to 3  do
begin
   for j:= 1 to 5 do
        write(c[ i , j ] ,'   ') ;
  writeln ;
end;
readln ;
end.
```

تمارين الفصـــل

✍ التمـــريـــن 01.11:

1. قدم التعليمة التي تسمح بالتصريح (Declaration) بجدول ثنائي البعد (مصفوفة) Tableau A Deux Dimensions من النوع **الصحيح** (Entier).
2. قدم طريقة كتابة (Syntaxe) التعليمة المتوصل إليها في السؤال الأول.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول ثنائي البعد (مصفوفة) من النوع **الصحيح** ذو البعد (5,8) أي خمسة أسطر و ثمانية أعمدة.

✍ التمـــريـــن 02.11:

1. قدم التعليمة التي تسمح بالتصريح (Declaration) بجدول ثنائي البعد (مصفوفة) Tableau Deux Dimensions من النوع **الحقيقي** صحيح (Réel).
2. قدم طريقة كتابة (Syntaxe) التعليمة المتوصل إليها في السؤال الأول.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول ثنائي البعد (مصفوفة) من النوع **الحقيقي** ذو البعد (10,7) أي عشرة أسطر و سبعة أعمدة.

✍ التمـــريـــن 03.11:

1. قدم التعليمة التي تسمح بالتصريح (Declaration) بجدول ثنائي البعد (مصفوفة) Tableau Deux Dimensions من النوع **أحرف** (caractères).
2. قدم طريقة كتابة (Syntaxe) التعليمة المتوصل إليها في السؤال الأول.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول ثنائي البعد (مصفوفة) من النوع **أحرف** ذو البعد (3,4) أي ثلاثة أسطر و أربعة أعمدة.

✍ التمـــريـــن 04.11:

1. قدم التعليمة التي تسمح بالتصريح (Declaration) بجدول ثنائي البعد (مصفوفة) Tableau Deux Dimensions من النوع **سلسلة أحرف** (Chaine de caractères).
2. قدم طريقة كتابة (Syntaxe) التعليمة المتوصل إليها في السؤال الأول.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول ثنائي البعد (مصفوفة) من النوع **سلسلة أحرف** ذو البعد (4,3) أي أربعة أسطر و ثلاثة أعمدة.

✍ التمــريــن 05.11:

1. أكتب البرنامج الذي يسمح بالتصريح بجدول ثنائي البعد (مصفوفة) من النوع **الصحيح** ذو البعد (m,n) أي m سطر و n عمود.
2. أكتب البرنامج الذي يسمح بالتصريح بجدول ثنائي البعد (مصفوفة) من النوع **الحقيقي** ذو البعد (m,n) أي m سطر و n عمود.
3. أكتب البرنامج الذي يسمح بالتصريح بجدول ثنائي البعد (مصفوفة) من النوع **أحرف** ذو البعد (m,n) أي m سطر و n عمود.
4. أكتب البرنامج الذي يسمح بالتصريح بجدول ثنائي البعد (مصفوفة) من النوع **سلسلة أحرف** ذو البعد (m,n) أي m سطر و n عمود.

✍ التمــريــن 06.11:

لتكن القيم التالية:

41	12	24	54
36	29	10	34
11	22	33	10
23	46	19	30
67	77	44	88

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بإدخال العناصر أعلاه عن طريق البرنامج (التوجيه المباشر) L'affèctation directe في جدول ثنائي البعد (مصفوفة) الذي يحمل الإسم *Mat* . أي أن العناصر تكون ضمن البرنامج.

✍ التمــريــن 07.11:

لتكن معطيات التمرين 06.11 أعلاه

- **المطلـوب:**

1. باستخدام التعليمة **for… To… do…** أكتب البرنامج الذي يسمح بإدخال العناصر أعلاه عن طريق لوحة المفاتيح الملامس (Le clavier) في جدول ثنائي البعد (مصفوفة).

✍ التمــريــن 08.11:

لتكن معطيات التمرين 06.11 أعلاه

- **المطلـوب:**

1. باستخدام التعليمة **while….do….** أكتب البرنامج الذي يسمح بإدخال العناصر أعلاه عن طريق لوحة المفاتيح الملامس (Le clavier) في جدول ثنائي البعد (مصفوفة).

✍ التمــريــن 09.11:

لتكن معطيات التمرين 06.11 أعلاه

- **المطلـوب:**

1. باستخدام التعليمة **repeat …. until ….** أكتب البرنامج الذي يسمح بإدخال العناصر أعلاه عن طريق لوحة المفاتيح الملامس (Le clavier) في جدول ثنائي البعد (مصفوفة).

✍ التمــريــن 10.11:

أكتب البرنامج الذي يسمح بإدخال عناصر مصفوفة ذات البعد (m,n) أي m سطر و n عمود في الحالات التالية:

1. باستخدام التعليمة **for… To… do…**

2. باستخدام التعليمة **while….do….**

3. باستخدام التعليمة **repeat …. until ….**

✍ التمــريــن 11.11:

لتكن المصفوفة *Matrice* ذات البعد $(3,5)$

$$Matrice = \begin{pmatrix} 23 & 13 & 78 & 90 & 43 \\ 56 & 12 & 76 & 33 & 44 \\ 32 & 88 & 91 & 67 & 55 \end{pmatrix}$$

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. بإدخال عناصر المصفوفة *Matrice*.

2. حساب مجموع عناصر المصفوفة *Matrice*.

3. نشر المجموع المتوصل إليه في السؤال 2 كما يلي:

La Somme Des Eléments De La Matrice Egale :

✍ التمــريــن 12.11:

لتكن المصفوفة *Matrice* المعطاة من خلال التمرين 10.11 أعلاه

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. بإدخال عناصر المصفوفة *Matrice* .
2. حساب مجموع عناصر الأعمدة الفردية للمصفوفة *Matrice* .
3. حساب مجموع عناصر الأعمدة الزوجية للمصفوفة *Matrice* .
4. حساب مجموع عناصر الأسطر الفردية للمصفوفة *Matrice* .
5. حساب مجموع عناصر الأسطر الزوجية للمصفوفة *Matrice* .
6. نشر النتائج المحسوبة كما يلي:

1. La Somme Des Eléments Des Colonnes Impaires Egale :
2. La Somme Des Eléments Des Colonnes paires Egale :
3. La Somme Des Eléments Des Lignes Impaires Egale :
4. La Somme Des Eléments Des Lignes paires Egale :

✍ التمــريــن 13.11:

أكتب البرنامج الذي يسمح بـ:

1. إدخال عناصر مصفوفة ذات البعد (m,n) أي m سطر و n عمود
2. حساب مجموع عناصر المصفوفة
3. حساب مجموع عناصر الأعمدة الفردية للمصفوفة.
4. حساب مجموع عناصر الأعمدة الزوجية للمصفوفة.
5. حساب مجموع عناصر الأسطر الفردية للمصفوفة.
6. حساب مجموع عناصر الأسطر الزوجية للمصفوفة.
7. حساب مجموع عناصر القطر.
8. نشر النتائج التي تم حسابها أعلاه.

✍ التمــريــن 14.11:

لتكن لدينا المصفوفتين *Mat_1* و *Mat_2* التاليين:

$$Mat_1 = \begin{pmatrix} 12 & 23 & 33 \\ 45 & 65 & 78 \\ 55 & 67 & 90 \\ 45 & 66 & 90 \\ 11 & 25 & 69 \end{pmatrix} \quad , \quad Mat_2 = \begin{pmatrix} 45 & 55 & 67 & 10 & 89 \\ 33 & 23 & 22 & 66 & 67 \\ 35 & 48 & 99 & 11 & 45 \\ 89 & 98 & 55 & 74 & 73 \\ 34 & 64 & 21 & 66 & 76 \end{pmatrix}$$

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال عناصر المصفوفة الأولى Mat_1.

2. إدخال عناصر المصفوفة الثانية Mat_2.

3. دمج (Fusion) المصفوفين Mat_1 و Mat_2 في مصفوفة ثالثة Mat_3 كما يلي:

$$Mat_3 = \begin{pmatrix} 12 & 23 & 33 & 45 & 55 & 67 & 10 & 89 \\ 45 & 65 & 78 & 33 & 23 & 22 & 66 & 67 \\ 55 & 67 & 90 & 35 & 48 & 99 & 11 & 45 \\ 45 & 66 & 90 & 89 & 98 & 55 & 74 & 73 \\ 11 & 25 & 69 & 34 & 64 & 21 & 66 & 76 \end{pmatrix}$$

✍ التمــريــن 15.11:

لتكن مصفوفة الوحدة A التالية:

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بتشكيل و تكوين مصفوفة الوحدة A ثم نشرها.

✍ التمــريــن 16.11:

لتكن لدينا المصفوفتين A و B التاليتين و التين لهما نفس البعد

$$A = \begin{pmatrix} 23 & 22 & 14 & 24 \\ 11 & 19 & 10 & 98 \\ 80 & 25 & 46 & 43 \\ 32 & 34 & 45 & 56 \\ 88 & 88 & 80 & 20 \end{pmatrix}_{(5,4)} \quad , \quad B = \begin{pmatrix} 56 & 90 & 76 & 51 \\ 56 & 21 & 90 & 00 \\ 78 & 87 & 98 & 90 \\ 51 & 10 & 74 & 12 \\ 89 & 87 & 90 & 11 \end{pmatrix}_{(5,4)}$$

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال عناصر المصفوفة الأولى A.
2. إدخال عناصر المصفوفة الثانية B.
3. حساب مجموع المصفوفتين A و B في مصفوفة ثالثة C أي أن $C = A + B$
4. نشر المصفوفة C

✍ التمـريـن 17.11:

لتكن لدينا المصفوفتين MAT_1 و MAT_2 التاليتين حيث أن:

$$MAT_2_{(4,5)} = \begin{pmatrix} 10 & 20 & 30 & 40 & 50 \\ 0 & 8 & 6 & 4 & 2 \\ 1 & 4 & 5 & 10 & 12 \\ 2 & 3 & 15 & 20 & 2 \end{pmatrix} \quad , \quad MAT_1_{(3,4)} = \begin{pmatrix} 2 & 5 & 1 & 3 \\ 4 & 6 & 8 & 10 \\ 1 & 4 & 2 & 8 \end{pmatrix}$$

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. إدخال عناصر المصفوفة الأولى MAT_1.
2. إدخال عناصر المصفوفة الثانية MAT_2.
3. حساب جداء المصفوفتين MAT_1 و MAT_2 في مصفوفة ثالثة C أي أن $C = A \times B$
4. نشر المصفوفة C

✍ التمـريـن 18.11:

لتكن المصفوفة $MATRICE$ التالية:

$$MATRICE = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

- **المطلـوب:**

1. أكتب البرنامج الذي يسمح بتشكيل و تكوين المصفوفة الوحدة *MATRICE* ثم نشرها.

✍ التمــريــن 19.11:

لتكن لدينا الجدول التالي:

12	23	123	178	89	90	87	
56	456	678	32	8	99	00	
34	65	231	56	676	678	65	
44	55	66	77	22	12	450	
65	87	90	321	56	778	887	

- **المطلـوب:**

أكتب البرنامج الذي يسمح بـ:

1. حساب مجموع عناصر كل سطر من أسطر الجدول و إدخال النتيجة في الخانة المقابلة للعمود الأخير.

2. حساب مجموع عناصر كل عمود من أعمدة الجدول و إدخال النتيجة في الخانة المقابلة للسطر الأخير.

المراجـع

I-/ المراجـع باللغة الأجنبيـة:

1. Anne-Marie DUSSAIX, Jean-Pierre INDJEHAGOPIAN, Méthodes statistique appliquées a la gestion, Les éditions d'organisations, Paris, 1981.
2. Belaid M.C, Algorithme et programmation en langage pascal : cours, exercices et travaux pratique avec corrigé, Les pages bleues internationales, Algérie, 2006
3. Benyakhlef Mustafa, Probabilités et statistique mathématique, Maroc, 1977.
4. Bernard Grais, Méthodes statistique, Dunod, Paris,
5. Calot G., Cours de calcul de probabilité, Dunod, Paris, 1971.
6. Calot G., Cours de statistique déscriptive, Dunod, Paris, 1973
7. Dominique Maille, Programmation en langage pascal, BERTTI Edition, Algérie, 1992.
8. Gilles Ouellet, Statistiques :Théorie Exemples Problèmes, Les éditions Le Griffon d'argile, Canada, 1985.
9. Guitton H., Statistique, Dalloz, Paris, 1971.
10. Delannoy Claude, Programmation en Turbo-Pascal, Edition Eyrolles, 1993.
11. Delannoy Claude, S'initier à la programmation ; Avec des exemples en C, C++, Edition Eyrolles, 2001.
12. Delannoy Claude, Programmer en langage C : Cours et exercices corrigés, Edition Eyrolles, 2014.
13. François Gervais, Turbo-Pascal 6 , 1991
14. Jacques Boisgontier et Christophe Donay, Turbo-Pascal et ses fichier – version 4 et 5, 1989
15. Murgulescu Michel, Algorithme et programmation en langage Turbo-Pascal, Edition Loze-Dion, 2002.
16. Morleghem J., Turbo-Pascal 4 Facile sur PC compalibles, 1988
17. Morleghem J., L'indispensable pour Turbo-Pascal sur PC version 3,4,5, Edition Marabout, 1990
18. Reiner Scholles, Le grand livre du Turbo et borland Pascal 7.0, 1993
19. Stephen O'Brien, Turbo-Pascal 7, Edition Dunod, France, 1991.