

Rappel

Structure d'un programme

Syntaxe:

```
PROGRAM nom_du_programme ;  
USES wincrt ;           { bibliothèque nécessaire pour read et write }  
CONST identificateur = valeur ;   { déclaration ( éventuelle ) de(s) constante(s) }  
VAR identificateur1: type1 ;  
      identificateur2: type2 ;     { déclaration des variables }  
BEGIN                               { début des instructions }  
  instruction_1;  
  instruction_2;  
  .....  
  instruction_n;  
END . { fin du programme }
```

Remarques:

Les instructions sont séparées par ' ; '

L'éditeur PASCAL ne distingue pas les majuscules des minuscules, ignore les tabulations, les répétitions d'espaces et les retours à la ligne.

Le ' END ' final est suivi d'un point. Tout ce qui suivra sera ignoré par le compilateur.

Ce qui est entre accolades { } est ignoré par le compilateur : Cela constitue un commentaire

Instructions élémentaires :

Entrée de données

Syntaxe **READLN (identificateur_de_variable);**

Permet l'entrée au clavier d'une variable

Seul un curseur clignotant indique que l'ordinateur attend une entrée. Il est judicieux d'afficher auparavant un message pour indiquer ce qu'on attend.

Affichage de données

Syntaxe

WRITE (identificateur_de_variable);

Affiche à l'écran, à partir de la position courante du curseur, le contenu (la valeur) de la variable

WRITE (donnee_1 , donnee_2 , ... , donnee_n);

Affiche à l'écran les données (contenu de variables, valeurs, ou textes) sur une même ligne, sans espace.

WRITELN (. . .); Même chose, mais passe à la ligne après l'affichage

Affectation

Syntaxe

identificateur_1 := identificateur_2 ;

Affecte à la variable 1 le contenu de la variable 2

identificateur := expression ;

Affecte à la variable 1 le résultat de l'expression

Exemples: A := B ; {assigne à A la valeur de B}

A := i/j ; {assigne à A le quotient (de type real) de i par j}

TP 2 INTRODUCTION A LA SYNTAXE PASCAL OPERATIONS ELEMENTAIRES

BUT DU TP : Manipulations et adaptations dans un environnement PASCAL (prendre connaissance du rôle de l'éditeur et du compilateur) en utilisant des structures de données de base et des opérations élémentaires.

A retenir :

- *readln* : saisie au clavier ; c'est l'équivalent de l'instruction *lire*
- *writeln* : affichage à l'écran ; c'est l'équivalent de l'instruction *écrire*
- « := » : affecter une variable ; c'est l'équivalent de ←
- Les différents types de variables simples en Pascal sont :
 - Integer* : nombre entier
 - Real* : nombre réel, la partie entière et la partie décimale sont séparées par un point « . »
 - Boolean* : booléen, ne peut prendre que les deux valeurs *true* ou *false*
 - Char* : caractèreactères
 - String* : chaine de car
- Les principaux opérateurs mathématiques en Pascal sont :
 - +*: addition
 - : soustraction
 - **: multiplication
 - /*: division (5 / 2 vaut 2.5)
 - div*: division entière (5 div 2 vaut 2)
 - mod*: reste de la division (5 mod 2 vaut 1)

Exercice 1 :

```
PROGRAM manip1;
USES crt ;
VAR
    a,b,c : integer;           { Partie de déclaration des variables }
                                { a, b et c sont 3 var. de type entier }
BEGIN
    Clrscr ;
    readln(a);                 { Saisie au clavier du contenu de a }
    b := 5;                    { Affectation avec le symbole « := » }
    c := a+b;
    writeln ('Valeur de c : ',c);
                                { Affichage d'un texte, puis du contenu }
                                { de c (séparé par une virgule) }
END.
```

1. Lancer l'éditeur de texte et taper le code source suivant :
2. Enregistrer dans un fichier *premier.pas* : tout fichier source en Pascal doit avoir l'extension « point pas ».
3. Après l'avoir compilé il faut exécuter le programme.

Que fait le programme ?

Exercice 2 :

```
PROGRAM
melange; VAR
x,y:real;
BEGIN
    write('x='); readln(x);
    write('y=');
    readln(y);
    x:=x+y;
    y:=x-y ; x:=x-y;
    writeln('après traitement :');
    writeln('x=',x, ',y=',y);
END.
```

1-Exécuter plusieurs fois ce programme et expliquer les résultats obtenus. que fait le programme.

2-Comment aurait-on pu faire autrement pour avoir la même action ?

Exercice 3: Soit le programme :

```

Program affectation ;
  Var a,b : integer ;
Begin
  b := 6 + 2 * 5 div 3 - 10; ----->
  a :=15 ; ----->
  b :=a ; ----->
  a :=a+1 ; ----->
  b :=b-3 ; ----->
  b :=a mod 3 ; ----->
end.
    
```

Valeurs des variables	
a	b

Inscrire dans les colonnes prévues , les valeurs prises par les variables successivement après l'exécution de chaque instruction. Les valeurs indéterminées seront notées par « ? ».

Exercice 4 :

Compléter et exécuter le programme qui demande à l'utilisateur les coordonnées de 2 points distincts A(xA,yA) et B(xB,yB) du plan et qui affiche les coordonnées du point milieu C de AB et la distance

$$AB = \sqrt{(xA - xB)^2 + (yA - yB)^2}$$

```

program milieu ;
  uses crt;
  var xA, yA, xB, yB, xC, yC,AB : ..... ;
begin
  writeln ('Entrez les coordonnées du premier point A' );
  ..... ;
  ..... ;
  writeln ('Entrez les coordonnées du deuxième point B' );
  ..... ;
  ..... ;
  xC := ..... ;
  yC := ..... ;
  writeln ('Les coordonnées du milieu sont :', ..... , ..... ) ;
  AB := ..... ;
  Writeln('la distance entre A et B est',.....) ;
end.
    
```

Exercice 5

Ecrire un programme Pascal qui permet de convertir et d'afficher en octets, kilo octets, méga octets et giga octets un nombre donné en **bits**.

Exercice 6

le programme Pascal suivant permet d'extraire puis afficher les chiffres de centaine c, dizaine d et unité u d'un entier x composé de trois chiffres.

```

Program EXTRAIRE;
  Uses Crt;
  Var x,c,d,u: integer;
Begin
  Writeln(' Donnez un entier composé de trois chiffres' );
  Readln(.....);
  c:= .....;
  d:= .....;
  u:= .....;
  Writeln(c, d, u);
End.
    
```

Exercice 7

Un nombre complexe $z=a+bi$ est caractérisé par sa partie réelle a et sa partie imaginaire b . Soient 2 nombres complexes $z_1=a_1+b_1i$ et $z_2=a_2+b_2i$, on a :

$$z_1+z_2=(a_1+a_2) + (b_1+b_2)i$$

$$z_1*z_2=(a_1*a_2 + b_1*b_2)+(a_1*b_2+a_2*b_1)i$$

compléter le programme qui saisit 2 nombres complexes et affiche leur somme et leur produit.

Program nombrecomplexe ;

Uses crt ;

Var a1,a2,b1,b2 : ;

Begin

Writeln ('donner la partie réelle de z1') ;

..... ;

Writeln ('donner la partie imaginaire de z1') ;

..... ;

Writeln ('donner la partie réelle de z2') ;

..... ;

Writeln ('donner la partie imaginaire de z2') ;

..... ;

Writeln('z1 + z2 =',, '+',, '* i');

Writeln('z1 * z2 =',, '+',, '* i');

End.