

## TP N° 3 : PASCAL (NOTIONS FONDAMENTAUX ET TURBO PASCAL)

### I- Introduction

1. Qu'est-ce qu'un programme ? Qu'est qu'un langage de programmation ?
2. Qu'est-ce qu'une application et quelle est sa relation avec le logiciel ?
3. Qu'elle est l'intérêt de passer par une analyse préalable avant de passer en écriture du programme? quelles sont ses différentes formes ?
4. Quelle est la différence entre les erreurs syntaxiques et les erreurs sémantiques ?

### II- Le Langage Pascal

1. Quelle est la structure générale d'un programme écrit en Pascal ?
2. Comment déclarer les constantes et les variables en Pascal ?
3. Comment faire entrer des données (lecture) et afficher des résultats (écriture) en Pascal ?
4. Quelle est la structure de l'instruction d'affectation en Pascal ? quels sont les différents opérateurs utilisés pour former une expression ? ces opérateurs ont-ils la même priorité d'évaluation ?

### III. Exercice 1

Les expressions suivantes sont-elles correctes ? Répondre **Correcte** ou **Incorrecte**.

- 1- Program programme ;
- 2- Program équation ;
- 3- Program affichage
- 4- Program pair impair ;
- 5- If (a :=3) then  
    c := a+b ; (\* sachant que a, b et c de type réel \*)
- 6- T :=(x=y) and (x<>z) ; (\* sachant que T de type booléen et x, y, z de type entier \*)
- 7- Read (A) ; (\* sachant que A de type entier \*)
- 8- Read ('A') ; (\* sachant que A de type caractère \*)
- 9- write (Ahmed) ; (\* sachant que Ahmed de type entier \*)
- 10- write ('Ahmed') ; (\* sachant que Ahmed de type caractère \*)
- 11- if (s='s') then  
    write ('Incorrect') ; (\* sachant que s de type caractère \*)
- 12- If (Ali=Omar) then            Ali := (a=b) ; (\* sachant que a, b de type entier et Ali et Omar de type booléen \*)

### IV- Exercice2

1. Ouvrir l'environnement Pascal et Créer un nouveau fichier
2. Écrire le programme « exo2 » qui apparaît dans la figure
3. Enregistrer le programme puis le Corriger.
4. Exécuter le programme et deviner ce que fait ?
5. Réenregistre le programme sur vos flache et quitter Pascal.

### V. Programme en pascal

```
PROGRAM TEST (input,output)
Var
  A,C,F : Integer ;
  D,E : Real ;
BEGIN
```

```
programme exo2.
variables a,b :entier;
x: réel;
begin
write('Entrer les coefficients').
read(a).
read(b);
if a=0 then
  if b=0 then
    write('infinite de solution');
  else
    write('pas de solution');
else
  begin
    x ← -b/a;
    write('la solution est:',x);
  end;
fin.
```

```

WRITELN('Opérations élémentaires avec différents types d'erreurs');
WRITELN('Donner les entiers A et C')
READLN A;
B :=20 ;
D :=A/B ;
E :=A/C ;
F :=A*B ;
WRITELN ('Les résultats obtenus sont :D,E,F);
END.

```

### Travail demandé :

1. Après la saisie puis la compilation du programme, donnez les erreurs syntaxiques et sémantiques
2. Après correction des erreurs, sauvegarder le programme sur disque, puis effectuer la compilation.
3. Exécutez le programme en donnant à la variable A la valeur : A=10. Que se passe-t-il et pourquoi ?
4. Corriger le programme puis l'exécuter de nouveau en donnant A=10 et C=10, puis A=10 et C=2 ?
5. Corriger le programme par insertion d'un point d'arrêt avant la dernière instruction du programme. Que remarquez-vous et pourquoi ?
6. Transformez la dernière ligne du programme en trois lignes d'affichage, chacune affichant une des variables D, E, F. Que remarquez-vous et pourquoi ?

### VI. Remarques importantes

<ul style="list-style-type: none"> <li>• Pour <b>bien répondre</b> aux questions de paragraphes I et II, il faut voir ou se référer au cours.</li> <li>• Le <b>PASCAL</b>, créé par <b>WIRTH</b> au début des années <b>70</b>, possède des instructions assez et favorise une approche méthodique et disciplinée (on dit "structurée").</li> <li>• Par défaut, lorsque l'on affiche un nombre réel avec <b>Write</b> ou <b>Writeln</b>, il est affiché sous forme scientifique, correspond à une mantisse réelle m à 20 chiffres (<math>1 \leq m &lt; 10</math>) et un exposant e signé à 4 chiffres. Ainsi le nombre 15 serait 1.5000000000000000E +0001. Pour l'écrire sous forme 15.0, on utilise le format suivant : <b>(15 :4 :2)</b>. qui affiche le nombre 15, sur un total de 4 caractères et avec <b>2</b> chiffres après la virgule.</li> <li>• Avant la <b>fin</b> du programme (càd l'instruction <b>END</b>. Il faut toujours rajouter l'instruction <b>readln</b>).</li> <li>• La <b>virgule décimale</b> est toujours représentée par un <b>point</b> en informatique.</li> <li>• Un identificateur (tout nom que vous choisissez : variable, programme...) peut être formé de lettres (A à Z), de chiffres du signe <b>_</b> (souligné).</li> <li>• Le premier caractère doit être une lettre. Par exemple, VALEUR1 ou PREM_VALEUR sont possibles mais pas IERE_VALEUR.</li> <li>• TURBO PASCAL accepte des noms de 127 caractères maximum.</li> </ul>	<ul style="list-style-type: none"> <li>• Les types simples connus en PASCAL sont : REAL, INTEGER (entier naturel), CHAR (contient UN est un seul caractère), STRING et BOOLEAN.</li> <li>• Toute <b>variable</b> utilisée dans un programme doit être <b>déclarée</b>. Ceci évite la plupart des erreurs de frappe, et rend le programme plus compréhensible.</li> <li>• En PASCAL les <b>minuscules</b> sont traitées comme des <b>majuscules</b> (SURface et surFACE désignent la même case mémoire).</li> <li>• Les <b>accents</b> et autres <b>ç</b> ne sont <b>pas autorisés</b>.</li> <li>• Un <b>blanc</b> dans un identificateur est également <b>interdit</b> (utilisez <b>_</b> pour séparer des mots dans un même identificateur).</li> <li>• Les textes doivent être entourés de cotes ('). Les majuscules/minuscules sont significatives. Pour afficher une apostrophe utiliser deux cotes ('"exemple"). Pour sauter une ligne utiliser WRITELN seul.</li> <li>• Les instructions doivent toujours être séparées par des ";" (j'ai dit "séparées", pas "terminées").</li> <li>• On peut insérer des remarques dans le programme (qui ne seront pas lues par le compilateur) en les entourant par (* et *).</li> <li>• Le programme doit toujours se terminer par un <b>point</b>.</li> </ul>
---	---

