

## I) Introduction:

Dans ce chapitre on va voir comment s'y prendre quand on a un problème à résoudre avec un ordinateur. En partant du premier énoncé du problème jusqu'au produit fini de l'ordinateur, quelles sont les étapes à suivre ?

## II) Les étapes de résolution d'un problème :

**1) Exemple 1:** *Faites une application qui permet de calculer le bénéfice qu'un commerçant peut faire sur la vente d'un produit donné.*

⇒ Le problème est posé par le commerçant. On l'appelle l'instigateur du problème.

Quelles seront les différentes étapes à faire par l'informaticien pour arriver à une solution pour ce commerçant?

### 2) Spécifier en détails le problème :

Il s'agit de :

- Bien lire l'énoncé du problème et le comprendre.
- Spécifier avec le plus de précision les données à utiliser dans l'application et le résultat auquel on veut arriver tout en évoquant les formulations de calcul nécessaires.

Le but du problème est d'afficher sur l'écran le bénéfice fait sur la vente d'un produit donné.

⇒ Le bénéfice est égal au prix de vente auquel on enlève le prix de revient.

⇒ Le prix de revient est égal à la somme du prix d'achat et des frais.

⇒ Le prix d'achat, les frais et le prix de vente sont trois données.

### 3) analyse du problème :

Il existe plusieurs approches pour analyser un problème. Par exemple, l'analyse ascendante et l'analyse descendante.

✓ L'analyse ascendante : consiste à partir du niveau le plus bas (données) vers le niveau le plus haut (résultat).

✓ L'analyse descendante : consiste à partir du niveau le plus haut (résultat) vers le niveau le plus bas (données).

**Remarque :** L'analyse qu'on va utiliser pour la résolution des problèmes durant cette année est l'analyse descendante.

Pour établir cette analyse, on va dresser une grille comportant le nom du programme ou de l'activité, la liste des définitions de tous les objets et toutes les actions qui paraîtront dans l'élaboration de la solution de cette application.

⇒ Cette liste commence impérativement du résultat. Ce dernier doit être correctement défini.

⇒ Tout autre élément nouveau doit être à son tour défini.



NOM : CALCUL BENEFICE		
SEQUENCES	L.D.E. (liste des définitions explicites)	O.U.(objets utilisés)
6	Résultat = écrire (bénéfice)	
5	Bénéfice $\leftarrow$ prix vente - prix revient	Bénéfice
4	Prix vente = donnée	Prix de vente
3	Prix de revient = prix d'achat + frais	Prix de revient
1	Prix d'achat = données	Prix d'achat
2	Frais = données	frais
7	FIN < CALCUL BENEFICE >	

⇒ Nous avons utilisé le verbe Ecrire pour traduire l'action d'affichage, on mettra entre parenthèses les objets à afficher ou dont on veut afficher le contenu.

Le verbe Ecrire n'est plus à définir, car il est un mot réservé à nous pour signaler un affichage.

⇒ On continue à définir les objets mentionnés dans la colonne des objets jusqu'à ce qu'on aura défini tous les éléments de la grille.

⇒ Dans la colonne séquence on doit établir l'ordre dans lequel les actions ou instructions seront exécutés.

#### **4) Elaboration de l'algorithme :**

L'écriture dans l'ordre, des différentes actions, donne une suite finie d'actions appelée algorithme.

**Définition** : Un algorithme est une suite structurée et finie d'actions ou d'instructions pour résoudre un problème.

Ecrivons l'algorithme déduit de notre grille d'analyse.

- 0) Début CALCUL BENEFICE
- 1) Lire (prix d'achat)
- 2) Lire (frais)
- 3) Prix de revient  $\leftarrow$  prix d'achat + frais
- 4) Lire (prix vente)
- 5) Bénéfice  $\leftarrow$  prix vente – prix de revient
- 6) Ecrire (bénéfice)
- 7) Fin CALCUL BENEFICE

## 5) Traduction en programme:

L'ordinateur va faire l'exécution de cet algorithme. Seulement, nous devons écrire cet algorithme dans un langage que l'ordinateur comprendra. Nous aurions aimé que ce dernier comprenne notre langage naturel et le travail serait réduit et facilité. L'ordinateur est une machine électronique qui ne manipule que des impulsions électriques (0 et 1).

| C'est avec un langage de programmation qu'on va traduire les algorithmes.

### a / Caractéristiques d'un langage de programmation :

Un langage de programmation sert à écrire un algorithme en un programme traduisible en langage binaire ou encore langage machine pour que l'ordinateur puisse l'exécuter.

Il est composé d'un vocabulaire obéissant à une structure prédéfinie.

### b / Pourquoi ne pas utiliser le langage naturel ?

Contrairement à notre langage quel qu'il soit, l'Arabe, le Français ou l'Anglais, un langage de programmation doit être très précis. Il est composé de mots et d'expressions qui ne prêtent pas à confusion. En effet, un mot dans le langage habituel peut avoir plusieurs significations exemple MONTRE, alors qu'en langage de programmation chaque mot a un seul sens.

### c / exemple de langages de programmations :

Algol, c++, cobol, fortran, java, pascal, prolog.....

#### Remarques :

| Pour programmer en un langage donné, il faut bien le connaître.

| Pour écrire et exécuter un programme en un langage de programmation, on utilise l'éditeur de texte de ce dernier.

### d / L'interpréteur : (interprétation)

C'est un programme du logiciel de programmation qui analyse et contrôle ( traduit en langage machine) chaque instruction d'un programme et qui exécute immédiatement ce qui est demandé. S'il y a une erreur dans le texte du programme l'exécution s'arrête.



### e / Le compilateur : (compilation)

C'est un programme du logiciel de programmation qui analyse chaque ligne d'instruction de notre programme et qui traduit chaque instruction du langage évolué en langage machine.

Le programme compilé s'appelle programme objet. Il peut être enregistré sur une mémoire de masse sous forme d'un programme exécutable.



### f / L'exécuteur : (exécution)

C'est un programme du logiciel de programmation qui compile tout le programme et l'exécute instruction par instruction.



**Remarque :**

Il est toujours utile de faire des tests adéquats sur le programme.

Sachez que :

*Un programme fait ce qu'on lui a fourni et non ce qu'on pense lui avoir fournir.*

Pensez à cette remarque et vérifiez que le programme fait bien ce que vous pensez lui avoir demandé de faire.

**III) passage de l'algorithme au programme :**

Prenons le même exemple et essayons de traduire l'algorithme en programme pascal et l'exécuter sur machine.

**Algorithme**

- 8) Début CALCUL BENEFICE
- 9) Lire (prix d'achat)
- 10) Lire (frais)
- 11) Prix de revient ← prix d'achat + frais
- 12) Lire (prix vente)
- 13) Bénéfice ← prix vente – prix de revient
- 14) Ecrire (bénéfice)
- 7) Fin CALCUL BENEFICE

**programme (écrit en pascal)**

```
program CALCUL_BENEFICE;  
uses wincrt;  
var  
    b , pv , pr , pa , f : real;  
begin  
    read ( pa );  
    read ( f );  
    read ( pv );  
  
    pr := pa + f;  
    b := pv - pr;  
    write ( b );  
  
End.
```