

## Chapitre 5

# Démarche de Résolution de Problèmes

## Leçon

### Les étapes de résolution d'un problème

#### Activité :

Nous avons vu dans les chapitres précédents qu'il y a plusieurs programmes nous ont servis pour réaliser différentes tâches.

Exemple :

- ✗ Le programme nécessaire pour aux pages Web : → Internet Explorer
- ✗ Le programme nécessaire pour démarrer et exploiter l'ordinateur : → Le système D'exploitation.
- ✗ Le programme nécessaire pour traiter les textes : → MS-WORD

#### I. Introduction :

L'informatique et l'utilisation des ordinateurs deviennent nécessaires dans les différents domaines de la vie courante.

- \* Dans les pharmacies, il faut avoir un programme pour gérer les médicaments.
- \* Pour gérer les transactions d'achats et de ventes d'un magasin, il faut avoir un programme de gestion.
- \* Dans les bureaux de poste, il faut avoir un programme de gestion pour les actions financières réalisées (virement, le règlement des factures,....

→ Toute activité nécessite l'utilisation d'un **programme** qui sera **exécuté par l'ordinateur**.

**II. Etapes de résolution d'un problème :**

La réalisation d'un programme exécutable par un ordinateur, nécessite le suivi d'une démarche constituée d'un ensemble d'étapes.

**1. Première étape : Position du problème :**

Le problème est souvent posé par un demandeur de solution informatique. C'est le cas du pharmacien, d'un élève, d'un banquier,...

Parfois, ces demandeurs ne savent plus exprimer leurs besoins avec précision. L'objectif de cette étape est bien formuler le problème pour pouvoir le résoudre correctement.

**Activité :** Cet élève rencontre le problème suivant ?



Comment je peux convertir 4500 m  
en centimètres ??????????

**Q :** Aider cet élève à exprimer son problème ?

**R :** Le problème est : « *Ecrire un programme qui permet de donner une distance en mètres et d'avoir son équivalente en centimètres* ».

**2. Deuxième étape : Spécification et analyse des problèmes :**

L'objectif de cette étape est bien comprendre l'énoncé du problème, déterminer les formules de calculs, les règles de gestion,...

L'analyse des problèmes s'intéresse aux éléments suivants :

- \* Les résultats souhaités (sorties),
- \* Les traitements (actions réalisés pour atteindre le résultat),
- \* Les données nécessaires aux traitements (entrées).

Résultat = .....

Traitement = .....

Données : .....

L'analyse d'un problème est réalisée en suivant une approche d'analyse :

<b>L'approche ascendante</b>	<b>L'approche descendante</b>
Commence l'analyse à partir de données, ensuite mettre le traitement nécessaire, jusqu'à arriver au résultat.	Commence l'analyse à partir du résultat du problème, mettre le traitement nécessaire et finalement les données.
<b>Le problème de l'activité</b>	
<u>Les données :</u> Distance en mètres (DM) <u>Le traitement :</u> Conversion de la distance $DCM \leftarrow DM * 100$ <u>Le Résultat :</u> Afficher (DCM)	Résultat = Afficher (DCM) Traitement = $DCM \leftarrow DM * 100$ Donnée : DM

→ L'approche que nous allons suivre est l'approche descendante.

**L'analyse du problème de la conversion :**

Résultat = Ecrire (DCM)  
 Traitement =  
 $DCM \leftarrow DM * 100$   
 Donnée : DM

**3. Troisième étape : Ecriture de l'algorithme.**

Après avoir terminé l'analyse, il faut mettre les instructions dans leur ordre logique d'exécution. On obtient un algorithme.

« Un algorithme est une suite structurée et finie d'actions ou d'instructions permettant de résoudre un problème ».

**L'algorithme relatif a la conversion de la distance :**

- 0) Début conversion
- 1) Lire (DM)
- 2)  $DCM \leftarrow DM * 100$
- 3) Ecrire (DCM)
- 4) Fin conversion.

Un algorithme utilise les conventions suivantes :

- Le verbe « Lire » utilisé pour la saisie des données.
- Le verbe « Ecrire » utilisé pour l'affichage du résultat.
- Le signe «  $\leftarrow$  » utilisé pour mettre une valeur dans une variable.

**4. Quatrième étape : Ecriture du programme.**

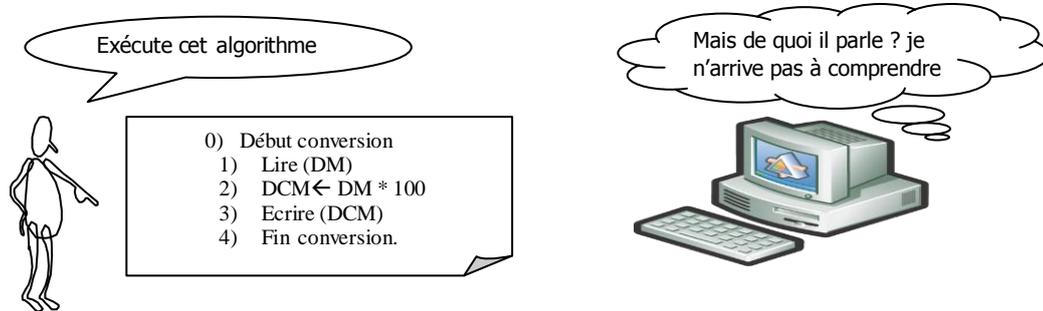
- Une fois l'algorithme du problème est établi, on doit penser à son exécution par l'ordinateur. Mais pensez-vous qu'un algorithme tel que vous avez rédigé, soit directement **utilisable** sur un ordinateur ? **Non.**

→ Pourquoi ?

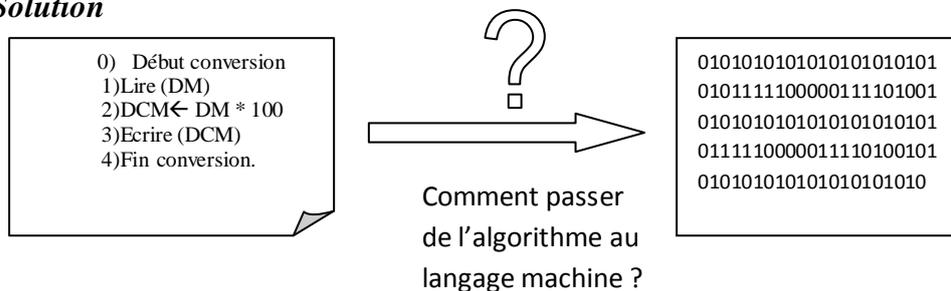
- Un algorithme est rédigé en langage humain qui n'est pas compréhensible par l'ordinateur.

→ **Constatation**

- Nécessité de transformer l'algorithme en un code **compréhensible** par l'ordinateur, c'est à dire en **langage machine**. Il est constitué de « 0 » et « 1 » et c'est le seul langage « compris » et exécuté par l'ordinateur.



→ **Solution**



→ Il faut traduire l'algorithme à l'aide d'un langage de programmation.

**a. Langage de programmation :**

- Un langage de programmation sert à la traduction d'un algorithme en un programme source compréhensible par l'ordinateur. Il est composé de mots clé, de symboles, obéissant à des règles de **syntaxe** (la façon d'écrire une instruction) et de sémantique (sens du l'instruction).

**Exemple :** PASCAL, JAVA, FORTRAN, COBOL, BASIC,.....

**b. Passage de l'algorithme au programme :**

Pour la traduction, on va utiliser le langage de programmation Pascal.

**L'algorithme**

0) Début conversion  
1) Lire (DM)  
2)  $DCM \leftarrow DM * 100$   
3) Ecrire (DCM)  
4) Fin conversion.

**Le code source**

```
program conversion ;  
uses winCRT ;  
var dm,dcm : real ;  
begin  
writeln('donner la distance en mètre') ;  
readln(dm) ;  
dcm :=dm*100 ;  
writeln('la distance en centimetre est',dcm) ;  
end.
```



Q : Est-ce que le code source obtenu peut être **exécuté** par l'ordinateur ?

R : Non. Pourquoi ?

R : Ce n'est pas du **langage machine**.

Q : Que faut-il faire donc ?

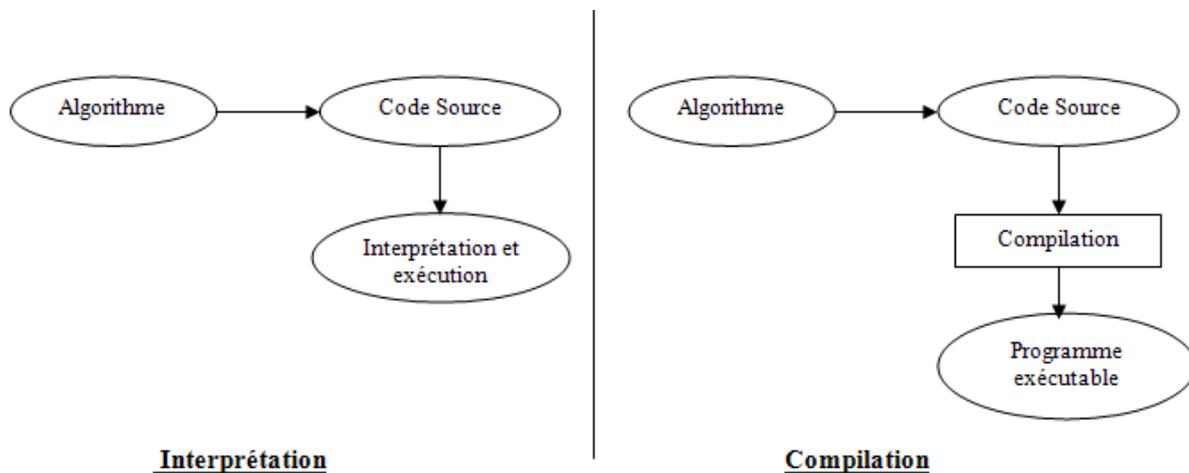
R : Transformer le **code source** en **langage binaire**.

→ Cette étape peut être appelée **Interprétation** ou **Compilation** selon le type du langage utilisé.

**c. Les types de langages de programmation :**

✘ Un langage de programmation est dit **interprété** si le code source sera traduit en langage machine et exécuté instruction par instruction.

✘ Un langage de programmation est dit **compilé** si tout le code source sera traduit en langage machine avant l'exécution. Cette tâche est assurée par le compilateur.



✘ Pascal est un langage de programmation compilé.

**Remarque :**

- Comme on a vu dans le problème précédent, un programme manipule des données en entrée, au cours du traitement ou en sortie.
- Ces données nécessitent un emplacement en mémoire. C'est pour cela, on doit préciser leurs types pour qu'ils prennent les emplacements adéquats.
- Au moment de l'analyse donc, il faut dresser un Tableau de Déclaration des Objets TDO).

Objet	Type/Nature	Rôle

**Activité :**

Dresser le TDO de l'algorithme « Conversion ».

**Solution :**

Objet	Type/Nature	Rôle
DM	Réel	Distance en mètres
DCM	Réel	Distance en Centimètres.

**Cinquième étape : Exécutions et test du programme.**

Une fois compilé ou interprété, un programme doit être testé pour s'assurer de son fonctionnement et qu'il répond aux besoins exprimés par l'utilisateur.

Un programme est testé par un jeu de test (des valeurs différentes de données).