

Question 3	La résolution de tous les problèmes de gestion est-elle automatisable ?	
Chapitre 3.1	<i>Introduction à la programmation</i>	Bac STMG SIG
Cours		Lycée J. Feyder

Chapitre 3.1

Introduction à la programmation

Question 3	La résolution de tous les problèmes de gestion est-elle automatisable ?	
Chapitre 3.1	<i>Introduction à la programmation</i>	Bac STMG SIG
Cours		Lycée J. Feyder

1 LA DEMARCHE DE PROGRAMMATION

L'écriture d'un programme répond à une démarche algorithmique, c'est-à-dire une démarche de résolution de problème.

Un programme va mettre en œuvre la logique suivante :



Pour aboutir à cela, le programmeur mène, lui, une démarche dans un sens différent :

1. Quel est le résultat que je souhaite voir délivrer par le programme ?
2. Quelles sont les données que je peux mettre à la disposition du programme (saisie, interrogation d'une base de données) ?
3. Quels traitements dois-je programmer pour passer de ces données à ces résultats ?

Exemple : Calcul d'un prix TTC à partir d'un prix HT

Etape	Mise en oeuvre	Exemple
1. Entrée	Somme : saisie au clavier	1000
2. Traitement	Somme * 1,196	1000 * 1,196
3. Résultat	Affichage de la somme TTC	1196

Une fois le raisonnement ainsi déterminé, le programmeur peut le coder dans un langage de programmation.

Question 3	La résolution de tous les problèmes de gestion est-elle automatisable ?	
Chapitre 3.1	<i>Introduction à la programmation</i>	Bac STMG SIG
Cours		Lycée J. Feyder

2 LES LANGAGES DE PROGRAMMATION

Un langage de programmation propose des instructions, proches du langage humain, qui permettent de définir des traitements. Ces instructions seront ensuite automatiquement « traduites » en langage machine exécutable par le processeur de l'ordinateur.

On distingue deux types de langage :

- **Interprété** : les instructions sont interprétées, au fur et à mesure de leur exécution, par un programme intermédiaire appelé interpréteur,
- **Compilé** : un programme annexe, appelé compilateur, donne naissance à un fichier exécutable (en langage machine) de façon autonome (sans avoir recours à nouveau au compilateur).

Par exemple PHP est un langage interprété, son code sera obligatoirement exécuté par un serveur web.

C est un langage compilé, le programme sera exécutable sans programme annexe.

3 LES VARIABLES

Les programmes manipulent des valeurs (saisies, calculées) qui sont stockées dans des **variables**, c'est-à-dire des emplacements dans la mémoire vive de l'ordinateur (et, de ce fait, l'emplacement est libéré en fin d'exécution du programme et la valeur perdue).

Une variable peut changer de valeur au cours de l'exécution d'un programme, chaque nouvelle valeur remplaçant (« écrasant ») la précédente. Par prudence, on doit considérer qu'une variable n'a pas de valeur par défaut en début de programme (même si de nombreux langages de programmation en attribuent une).

Il est à noter qu'une donnée dont la valeur ne change pas d'une exécution à l'autre du programme doit plutôt être enregistrée en **constante** afin d'éviter inutilement sa ressaisie.

Une variable ou une constante est caractérisée par :

- un nom (sans caractères spéciaux, sans espace et sans commencer par un chiffre),
- un type de données (texte, numérique, date, etc...).

Question 3	La résolution de tous les problèmes de gestion est-elle automatisable ?	
Chapitre 3.1	<i>Introduction à la programmation</i>	Bac STMG SIG
Cours		Lycée J. Feyder

4 L'AFFECTATION

L'opération d'affectation permet d'attribuer une valeur (ou le résultat de l'évaluation d'une expression arithmétique ou logique) à une variable. Elle s'effectue avec l'opérateur '='.

Syntaxe d'affectation d'une variable

nomVariable = valeur

Le principe de fonctionnement est le suivant : le processeur évalue l'expression arithmétique ou logique à droite du '=', puis stocke le résultat de cette évaluation dans la variable située à gauche du signe '='.

age = 18	affectation d'une valeur numérique
majeur =True	affectation d'une valeur booléenne
prenom = "Leon"	affectation d'une chaîne de caractères (entre guillemets)
TTC = HT +TVA	affectation du résultat d'une expression arithmétique

5 LES ERREURS DE PROGRAMMATION

Le **débogage** sert au programmeur à rechercher ses erreurs dans le programme.

On distingue 4 types d'erreurs :

- de syntaxe,
- de compilation (par exemple oubli de déclaration d'une variable),
- d'exécution (demande d'un traitement impossible, par exemple une division par 0),
- de logique (le raisonnement n'est pas correct et le résultat obtenu est donc faux).