Fomesoutra.com
ça soutra !
Docs à portée de main

ESATIC ECOLE SUPERIEURE AFRICAINE DES TIC	Année scolaire 2014 - 2015 PROF : KONAN HYACINTHE LICENCE 1

INTRODUCTION A L'ALGORITHMIQUE

CHAPITRE 1 :
NOTION DE BASE

1. Généralités
2. Définition d'un Algorithme
3. Exemple
4. Description des éléments d'un algorithme
   4.1. Les déclarations
      4.1.1. Variables
      4.1.2. Constantes
      4.1.3. Types
   4.2. Les instructions

1. Généralités

Ce cours s'adresse à de futurs utilisateurs éclairés des systèmes informatiques qui, à défaut d'écrire eux même leurs programmes, auront besoin de communiquer de manière précise avec des informaticiens et donc de connaître la discipline de l'intérieur. Ce cours est donc conçu comme une initiation à l'algorithmique et à la programmation.

Avant d'écrire les programmes qui résoudront les problèmes, nous écrirons des algorithmes.

2. Définition d'un Algorithme

Dans le sens général, un algorithme peut prendre de nombreuses formes :

- recette de cuisine
- notice de montage
- feuille de calcul d'impôt
- protocole expérimental ...

Nous retiendrons la définition suivante :

Un **ALGORITHME** la description, faite par un individu A, d'un enchaînement d'actions destinées à êtres exécutées par un individu B. Pour que B puisse exécuter sans erreur les tâches commandées par A, il faut que la description faite par A soit claire, complète, et sans ambiguïté.

Pour nous, l'individu B, celui qui exécute, doit pouvoir être remplacé par un ordinateur. Il faut alors que les consignes soient données dans un langage interprétable par l'ordinateur. L'ensemble des ces consignes, écrite dans un langage de programmation, s'appelle un **PROGRAMME**. Il est exécuté par l'ordinateur.

Comme il est difficile et peu souhaitable (pour des raisons qui seront développées plus tard) d'écrire directement un programme pour un ordinateur, on écrit des algorithmes, qui sont l'expression formalisée des solutions aux problèmes que l'on doit résoudre. Que veut dire formaliser ?

Par exemple au lieu de poser la question **"quel nombre faut-il ajouter à 7 pour obtenir 15"**, on peut formaliser la question sous la forme de l'équation **7 + x = 15**.

Voici l'exemple d'un algorithme qui ressemble à ce que vous écrirez dans quelques semaines.

3. Exemple

```
ALGORITHME Intérêts
/* calcule le nombre d'années à attendre pour obtenir une somme
donnée avec un dépôt annuel identique chaque année. Ces données sont
saisies au clavier par l'utilisateur. Les montants sont en euros */

/* Déclarations */
CONSTANTES
/* identificateur = valeur rôle */
      taux = 4.5 /* taux annuel */
VARIABLES
/* identificateur  : type rôle */
      depot : REEL /* montant du dépôt annuel */
      somme : REEL /* somme courante */
      sommeFinale: REEL /* somme souhaitée */
      nbAn : ENTIER /* nombre d'années */
      interet REEL /* montant de l'intérêt annuel */
/* Instructions */
DEBUT
1     somme ← 0   /* on part d'une somme nulle */
2     nbAn ← 0    /* on part d'un nombre d'années nul */
            /* entrée des données de l'utilisateur */
3     ECRIRE ("montant du dépôt annuel : ")
4     LIRE (depot)
5     ECRIRE ("montant de la somme souhaitée : ")
6     LIRE (sommeFinale)
      /* traitement */
7     TANTQUE (somme < sommeFinale) FAIRE
8           somme ← somme + depot     /* premier jour de l'année */
9           nbAn ← nbAn + 1
10          interet ← (somme * taux) / 100
11          somme ← somme + interet    /* dernier jour de l'année */
12    FINTANTQUE
      /* affichage des résultats */
13    ECRIRE ("il faudra attendre : ", nbAn, " pour recevoir : ", somme, "
francs")
FIN
```

On suppose que :

- le dépôt est identique chaque année
- le dépôt est effectué le premier jour de l'année
- l'intérêt est ajouté le dernier jour de l'année
- on s'intéresse à la somme possédée le dernier jour de l'année

Remarque : Normalement un algorithme ne comporte pas de numéros de lignes. Ceux qui ont été introduits dans cet exemple ne sont là que pour faciliter la simulation qui sera faite plus loin..

4. Description des éléments d'un algorithme

Un algorithme se compose de 2 parties :

- les **déclarations**
- les **instructions**

Dans chaque partie, des **COMMENTAIRE**S doivent faciliter la compréhension du lecteur.

4.1. Les déclarations

C'est la partie de déclaration des **constantes**, des **variables**, ... On pourrait la comparer à la mise en place du dispositif expérimental d'une expérience de chimie : préparer la verrerie adéquate pour recevoir les solutions, ...

4.1.1. Variables

Une **VARIABLE** sert à mémoriser une **VALEUR**. C'est une zone de mémoire, physique dans un ordinateur, virtuelle dans un algorithme. Sa valeur peut changer au cours du temps. Pour manipuler une variable, on lui donne un **NOM** : son **IDENTIFICATEUR**. Lorsqu'on utilise l'identificateur d'une variable, c'est en fait pour manipuler la valeur de la variable. Un identificateur doit commencer par une lettre, il se compose de lettres, de chiffres et/ou du tiret souligné. On pourra adopter la règle suivante pour les identificateurs de variables : elles sont composées de minuscules, sauf aux ruptures de mots qui seront en majuscule (ex : **sommeFinale**).

Une variable est aussi caractérisée par son **TYPE**, c'est à dire une contrainte sur les valeurs qu'elle peut prendre. Si on déclare une variable de type **ENTIER**, on ne pourra pas y mettre un **REEL**. Dans un algorithme il faut veiller à la compatibilité des types de variables et des valeurs qu'on met dans les variables. Dans une machine, le type d'une variable correspond à la place qui est réservée dans la mémoire pour ranger les futures valeurs. On conçoit qu'il faudra réserver une plus grande place pour un réel que pour un entier.

Le **rôle** (qui est un commentaire) est indispensable pour indiquer à quoi sert la variable à moins que celui-ci ne soit explicitement exprimé par l'identificateur.

4.1.2. Constantes

Une **CONSTANTE** correspond à une valeur littérale. A chaque fois que la constante apparaît dans un algorithme, elle est remplacée par sa valeur. On pourra adopter la règle suivante pour les identificateurs de constantes : elles sont composées de majuscule avec un tiret bas pour les ruptures de mots (ex : **TAUX_ANNUEL**).

4.1.3. Types

Au début on travaillera avec 4 types :

Type	En algorithmique	En machine
ENTIER	Elément de l'ensemble **Z**	élément d'un sous ensemble de Z
REEL	Elément de l'ensemble **R**	élément d'un sous ensemble de Q
CARACTERE	Signe ou lettre ou chiffre	Idem – délimiteur : '
CHAINE	Suite de caractères	Idem – délimiteur : "

Lorsqu'on veut manipuler directement des valeurs de chacun de ces types, il faut prendre certaines précautions :

- pour les entiers ou réels, pas de problème : **13** ; **15.78**
- pour les caractères (chaînes) il peut y avoir ambiguïté : en effet si j'écris **N**, je désigne l'identificateur d'une variable et non le caractère donc je dois écrire **'N'** pour désigner le caractère. Les valeurs de type caractère ou chaîne doivent être entourées par des apostrophes pour les distinguer des identificateurs.

4.2. Les instructions

Elles constituent la partie active de l'algorithme. Le chapitre suivant définira précisément ce que sont les instructions. Ici on ne présente qu'un exemple d'exécution de l'algorithme précédent. Les instructions de l'algorithme s'exécutent **SEQUENTIELLEMENT** au cours du temps et modifient certaines valeurs des variables. On peut comparer une instruction à un événement qui agit et modifie l'état du système sur lequel il agit. L'état du système est constitué par l'ensemble des valeurs des variables.
Chaque numéro de ligne fait référence à la ligne de l'algorithme. Il est convenu que, si la valeur d'une variable est modifiée par l'instruction courante, la nouvelle valeur de la variable est inscrite dans la case correspondante. L'exemple ci-dessous présente une exécution de l'algorithme pour laquelle l'utilisateur a donné

- 800 pour le dépôt
- 2600 pour la somme à obtenir

	depot	somme	sommeFinale	nbAn	interet	Ecran	Exprbool
1		0					
2				0			
3						Montant du dépôt annuel	
4	800.00					800	
5						Montant de la somme souhaitée	
6			2600.00			2600	
7							Vrai
8		800.00					
9				1			
10					36		
11		836.00					
12							
7							Vrai
8		1636.00					
9				2			
10					73.62		
11		1709.62					
12							
7							Vrai
8		2509.62					
9				3			
10					112.93		
11		2622.55					
12							
7							Faux
13						Il faudra attendre : 3 années et vous recevrez 2622.55 francs	

FIN DU CHAPITRE 1