
Les données

Algorithmique et Programmation en C

Chapitre 3 Karim Bouzoubaa

Objectifs

- Comprendre les concepts de base de la mémoire
- Devenir familier avec les types de données primitifs
- Connaître la représentation des données en mémoire
- Connaître quelques exemples de types de données composés

Les variables

- Variable : donnée utilisée dans un algorithme/programme (e.g. `nombre`, `carre`)
- Variable : emplacement dans la mémoire centrale de l'ordinateur
- Lors d'une affectation (`nombre = 5`): l'unité centrale se charge de mettre une valeur (en code binaire) dans l'emplacement mémoire correspondant
- Lorsque cette variable est utilisée pour affichage ou pour être évaluée dans une expression, l'unité centrale va récupérer sa valeur de la mémoire pour effectuer le traitement demandé
- Chaque variable possède :
 - un nom (qui l'identifie de manière unique dans le programme)
 - un type (entier, chaîne de caractères, réel, etc.)
 - une taille (en octets)
 - une valeur (selon le type)
 - Exemple, `nombre` porte le nom `nombre`, est de type entier, a une taille de 4 octets en mémoire centrale et possède la valeur 5

Les variables - Le nom

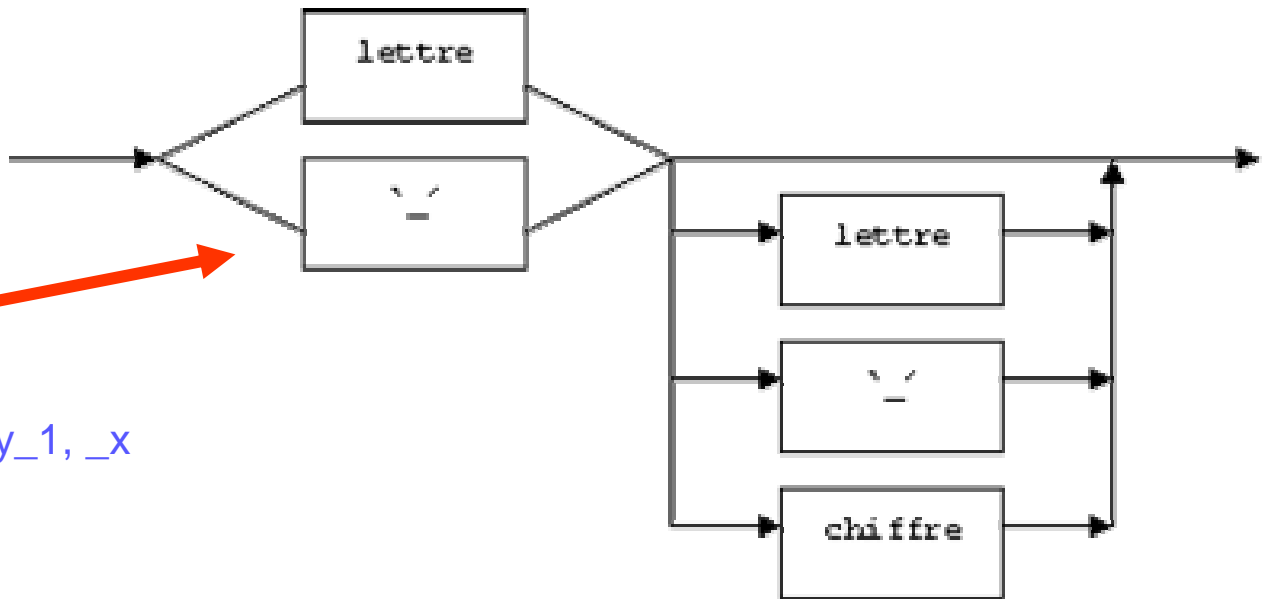
- Nommer une variable :

- Conditions

- Correct : x, x1, som, moy_1, _x
 - Non correct : 1x, x-y, x!

- Conventions

- Commencer une variable par une minuscule (e.g. nombre)
 - Si le nom de la variable est composé (de plusieurs mots), commencer chaque nouveau mot par une majuscule (e.g. monPremierNombre).



Les variables - Le type

- Le type d'une variable permet de connaître
 - sa taille en mémoire (en octets)
 - l'ensemble des valeurs possibles
 - les opérations que cette variable est susceptible de subir
- Types simples et types composés:
 - Un type est simple lorsque ses valeurs ne sont pas décomposables en constituants plus simples. Quatre catégories:
 1. Types entiers : qui correspondent aux nombres entiers positifs, nuls ou négatifs
 2. Types à virgule flottante : qui représentent les nombres avec une précision fractionnelle (voir rep. en mémoire)
 3. Type booléen : qui permet de représenter les valeurs logiques (TRUE/FALSE, 0/1)
 4. Type caractère :
 1. Les caractères alphabétiques: 'a' 'b' 'c' 'd' ... 'z' 'A' 'B' ... 'Z'
 2. Les caractères numériques '0' '1' ... '9'
 3. L'espace ' '
 4. Les caractères spéciaux '(' '-' '?' '*' ...

Les variables - Le type

- Types simples et types composés:
 - Un type est dit composé s'il est décomposable en d'autres types
 - Exemple: chaîne de caractères (String), décomposable en 0 ou plusieurs caractères
 - "salam"
 - "a4d? "

Les constantes

- Une constante est une donnée qui ne change de valeur durant tout le temps d'exécution du programme.
- Comme la variable, une constante porte
 - un nom,
 - un type,
 - une taille
 - et une valeur
- Pour spécifier qu'une donnée est constante, utiliser le mot clé `const`

```
const PI = 3.14 réel
```
- Les conventions d'écriture des constantes :
 - Tous les caractères du nom de la constante sont majuscules (e.g. `PI`)
 - Si le nom de la constante est composé (de plusieurs mots), séparer les mots par un souligné `'_'` (e.g. `MA_CONSTANTE`).