

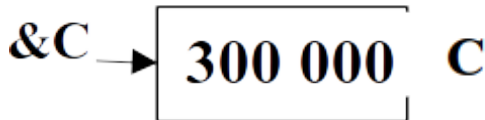


Algorithmique & Programmation en C

DECLARER CONSTANCE

Ressources :

[Diapos MEMORISER](#)



Objectif :

- Connaître la notion de *Constante*.
- Comment déclarer une constante dans un programme ?

1. DEFINITION

Il existe deux formes de **données** : les **variables** et les **constantes**. Les variables sont les plus couramment utilisées dans les programmes, car le programmeur peut changer leur valeur autant que nécessaire dans le programme. Les constantes sont, dans certains cas, préférées aux variables.

Exemples de constantes:

TVA= 0.196

PI= 3.14

Une **constante** est une **donnée** dont la **valeur ne peut pas varier en cours d'exécution du programme** : sa valeur (entière, réelle, caractère...) est fixée en début de programme, puis ne peut plus être changée par les instructions ; elles est **stockée dans la RAM en lecture seule** lorsque le programme est en cours d'exécution. Une constante, comme une variable, lors de l'exécution du programme, est décrite par :

- × un **nom** (étiquette associée à une zone mémoire RAM -ex. *C* sur le schéma en haut-) ;
- × un **valeur** (contenu *300 000* dans la zone mémoire sur le schéma) ;
- × une **adresse dans la RAM (pointeur &C** sur la zone RAM de la variable *C*).

Pour caractériser une constante, il faut lui associer un **type** de donnée, compatible avec sa nature :

- × sur combien de bits la valeur de la constante est-elle codée (par exemple, une **taille de sa zone mémoire** de 2 *octets* est associée à un type « petit entier naturel ») ?
- × la valeur de la constante est-elle négative (**type non signé** ou **type signé**) ?
- × quel est le **domaine de capacité** du type choisi (par exemple, un type « petit entier naturel » a des valeurs comprises dans $[0, 65535]$) ? **ATTENTION aux DEPASSEMENT de CAPACITE !!**
- × quels sont les opérateurs applicables à la constante (par exemple, + - / * et % pour des entiers) ?

Les types simples du langage C :

TYPE	Signification	CAPACITE	TAILLE (en octets)
char / unsigned char	caractère	$[-128, 127] / [0, 255]$	1
unsigned short	Entier non signé court	$[0, 65\ 535]$	2
short	Entier signé court	$[-32\ 768, 32\ 767]$	2
unsigned long ou unsigned int	Entier non signé long	$[0, 4\ 294\ 967\ 295]$	4
long ou int	Entier signé long	$[-2147483648, 2147483647]$	4
float	Réel simple précision	$[+/- 3.4 \cdot 10^{38}, +/- 3.4 \cdot 10^{38}]$	4
double	Réel double précision	$[+/- 1.7 \cdot 10^{308}, +/- 1.7 \cdot 10^{308}]$	8

2. ALGORITHME (DÉCLARATION CONSTANCE)

L'instruction de déclaration de constante permet d'indiquer les valeurs fixes que le programme pourra utiliser. Elle s'écrit dans la *partie données* de la fonction qui utilise cette constante :

```

/*****
                                DECLARATION DE CONSTANCE
*****/

ALGO Declarerconstante      // PARTIE DONNEES
    CONST NOMCONSTANTE= valeur : type // type numérique ou caractère
        PI= 3.14 : REEL simple précision // nombre PI pour calcul aire disque
DEBUT
FIN

```



Règle de programmation : nommage des constantes

- Noms explicites, pas trop longs ;
- TOUTES les lettres en majuscules ;
- Préciser rôle constante avec un commentaire explicatif.

Le programmeur choisira une donnée constante, plutôt qu'une variable, dans son algorithme lorsque la valeur de la donnée ne varie pas au cours de l'exécution du programme. Le **programmeur reconnaît visuellement qu'il s'agit d'une constante** (majuscules) et ne risquera pas de tenter de modifier sa valeur dans ses instructions.

Le programmeur choisira une donnée constante, plutôt qu'une valeur écrite directement dans les instructions, car le **nom explicite d'une constante dans une instruction est plus lisible qu'une valeur** : la signification de la grandeur est directement compréhensible.

D'autre part, lorsqu'une valeur constante est utilisée plusieurs fois dans un programme, il est **plus simple de la déclarer une seule fois pour la maintenance du programme**. Si cette valeur change un jour (comme par exemple pour un taux de TVA qui est constant lors de l'exécution du programme, mais qui peut varier à l'occasion d'un changement de gouvernement), le programmeur n'aura qu'à changer la valeur de sa constante, lors de sa déclaration, plutôt que toutes les instructions qui utilisent cette valeur.

Exemple d'utilisation de constante :

$$\text{prixTTC} = \text{prixHT} \times (1 + \text{TVA})$$

est préférable à

$$\text{prixTTC} = \text{prixHT} \times 1.196$$

