|  |  |  |
| --- | --- | --- |
| **séq. d'échapp.** | descr. anglaise | descr. française |
| **\n** | new line | passage à la ligne |
| **\t** | tabulator | tabulation |
| **\b** | **back** | curseur arrière |
| **\r** | return | retour au début de la ligne |
| **\"** | quotation marks | guillemets |
| **\\** | back-slash | trait oblique |
| **\0** | NUL | fin de chaîne |
| **\a** | attention (bell) | signal acoustique |

**[Previous ](http://www.ltam.lu/cours-c/solex05.htm)** **[Next ](http://www.ltam.lu/cours-c/solex07.htm)** **[Up ](http://www.ltam.lu/cours-c/solex01.htm)** **[Index ](http://www.ltam.lu/cours-c/solex.htm)** **[Contents ](http://www.ltam.lu/cours-c/solex_c.htm)**

**Exercice 2.5**

**#include <stdio.h>**

***/\* Ce programme calcule la somme de 4 nombres entiers***

***introduits au clavier.***

***\*/***

**main()**

**{**

**int NOMBRE, SOMME, COMPTEUR;**

***/\* Initialisation des variables \*/***

**SOMME = 0;**

**COMPTEUR = 0;**

***/\* Lecture des données \*/***

**while (COMPTEUR < 4)**

**{**

***/\* Lire la valeur du nombre suivant \*/***

**printf("Entrez un nombre entier :");**

**scanf("%i", &NOMBRE);**

***/\* Ajouter le nombre au résultat \*/***

**SOMME += NOMBRE;**

***/\* Incrémenter le compteur \*/***

**COMPTEUR++;**

**}**

***/\* Impression du résultat \*/***

**printf("La somme est: %i \n", SOMME);**

**return 0;**

**}**

**Solution:**

- Commande au compilateur: **#include<stdio.h>**

pour pouvoir utiliser les fonctions **printf** et **scanf**.

- Fonction **main**

n'a pas de paramètres (la liste des paramètres est vide) et

fournit par défaut un résultat du type **int** (à l'environnement).

- Commentaires (mis en italique).

- Variables utilisées:

NOMBRE, SOMME, COMPTEUR déclarées comme entiers (type **int**).

- Fonctions utilisées:

**printf**, **scanf** de la bibliothèque *<stdio>*.

- Opérateurs:

|  |  |  |  |
| --- | --- | --- | --- |
|  | += |  | opérateur arithmétique d'affectation |
|  | ++ |  | opérateur arithmétique |
|  | < |  | opérateur de comparaison |
|  | = |  | opérateur d'affectation |

- Structure de contrôle:**while(<condition>) { ... }**

répète le bloc d'instructions aussi longtemps que la <condition>  
est remplie.

- L'instruction **return 0;**

retourne la valeur zéro comme code d'erreur à l'environnement après  
l'exécution du programme

**Ajoutes :**

- la fonction **scanf**est appelée avec deux paramètres:

le format de saisie de la donnée (ici: "%i" pour lire un entier du type **int**)

l'adresse de la variable destination (ici: l'adresse de NOMBRE).

- la fonction **printf**est appelée avec un respectivement avec deux paramètres:

le premier paramètre est une chaîne de caractères, qui peut contenir une information pour le format d'affichage des variables indiquées dans la suite (ici: "%i" pour afficher la valeur du type **int** contenue dans SOMME).

les paramètres qui suivent la chaîne de caractères indiquent les noms des variables à afficher. (ici: SOMME)

### Exercice 4.3

Ecrire un programme qui lit un caractère au clavier et affiche le caractère ainsi que son code numérique :

**a)** en employant **getchar** et **printf**,

**#include <stdio.h>**

**main()**

**{**

**int C;**

**printf("Introduire un caractère suivi de 'Enter'\n");**

**C = getchar();**

**printf("Le caractère %c a le code ASCII %d\n", C, C);**

**return 0;**

**}**

**b)** en employant **getch** et **printf**.

**#include <stdio.h>**

**#include <conio.h>**

**main()**

**{**

**int C;**

**printf("Introduire un caractère \n");**

**C = getch();**

**printf("Le caractère %c a le code ASCII %d\n", C, C);**

**return 0;**

**}**

**#include <stdio.h>**

**main()**

**{**

**int A, B;**

**printf("Introduisez deux nombres entiers : ");**

**scanf("%i %i", &A, &B);**

**printf("Division entiere : %i\n", A/B);**

**printf("Reste : %i\n", A%B);**

**printf("Quotient rationnel : %f\n", (float)A/B);**

**return 0;**

**}**

**Remarque:***Conversion des types*

Pour obtenir un résultat rationnel, il faut qu'au moins l'une des deux opérandes soit d'un type rationnel. Si nous convertissons la valeur de l'opérande A en **float** à l'aide de l'opérateur de conversion forcée, alors la valeur de la deuxième opérande B est convertie automatiquement en **float**. Le résultat de la division est du type **float** et doit être représenté par le spécificateur de format **%f** (ou **%e**).

Remarquez bien que la conversion forcée concerne uniquement la valeur de la variable A ! La valeur de B est convertie automatiquement.

D'autre part, si nous écrivions:

**(float)(A/B)**

nous n'obtiendrions pas le résultat désiré, parce que la conversion se ferait seulement ***après***la division !