

---

## Travaux dirigés 4 : structure de contrôle *if* et *for*

---

L'objectif de ce TD est de vous familiariser avec la notion d'itération en programmation. On parle communément de "boucle". Cette notion sera illustrée sur des problèmes de comptage et de répétition d'actions.

### 1 Execution conditionnelle d'instructions : *if*

#### 1.1 Exercice type : Le minimum de 3 valeurs

Soient 3 variables *a*, *b*, *c*, initialisées à des valeurs quelconques. Écrire un programme qui calcule et affiche à l'écran le minimum des 3 valeurs.

### 2 Itération : l'instruction *for*

#### 2.1 Exercice type : calcul de $\sum_{i=1}^n i$

Écrire un programme qui calcule et affiche la somme des entiers de 1 à *n* :  $\sum_{i=1}^n i$ . *n* est un entier quelconque.

#### 2.2 La suite de Fibonacci

La *suite de Fibonacci* est une suite d'entiers telle que :

1. les deux premiers termes de la suite sont 1 et 1 ;
2. chaque terme suivant est la somme des deux termes qui le précèdent.

Tout nombre de cette suite est dit un *nombre de Fibonacci*. Le premier nombre de Fibonacci est donc 1, le deuxième est 1, le troisième est  $1 + 1 = 2$ , le quatrième est  $2 + 1 = 3$ , etc. Voici ci après un tableau avec les premiers dix nombres de Fibonacci.

<i>ordre</i>	1	2	3	4	5	6	7	8	9	10
<i>somme</i>	1	1	$1 + 1$	$2 + 1$	$3 + 2$	$5 + 3$	$8 + 5$	$13 + 8$	$21 + 13$	$44 + 21$
<i>nombre Fib.</i>	1	1	2	3	5	8	13	21	44	65

Écrire un programme qui affiche les 20 premiers nombres de Fibonacci.

### 3 Évaluation d'expressions booléennes

Soit le programme suivant :

```
#include <stdlib.h> /* EXIT_SUCCESS */
#include <stdio.h> /* printf */

#define FALSE 0
#define TRUE 1

/* Declaration de fonctions utilisateurs */
```

```

int main()
{
    int beau_temps = TRUE;
    int pas_de_vent = FALSE;

    printf("%d\n",beau_temps && pas_de_vent);
    printf("%d\n",beau_temps || pas_de_vent);
    printf("%d\n",! beau_temps || pas_de_vent);
    printf("%d\n",! (! beau_temps || pas_de_vent) == (beau_temps && ! pas_de_vent));

    return EXIT_SUCCESS;
}

```

/\* Definition de fonctions utilisateurs \*/

1. Qu'affiche le programme ?
2. Modifiez le programme pour qu'il demande la valeur des booléens à l'utilisateur (0 pour FALSE, sinon TRUE).

1. Que fait le programme suivant ?

```

1  /* declaration de fonctionnalites supplementaires */
2  #include <stdlib.h> /* EXIT_SUCCESS */
3  #include <stdio.h> /* printf, scanf */
4
5  /* declarations des constantes et types utilisateurs */
6
7  /* declarations des fonctions utilisateurs */
8
9  /* fonction principale */
10 int main()
11 {
12     int a;
13     double b;
14     char c;
15
16     printf("Entrez un nombre entier puis un nombre réel puis un caractère : ");
17
18     scanf("%d",&a);
19     scanf("%lg",&b);
20     scanf(" %c",&c);
21
22     printf("Vous avez saisi %d puis %g puis %c.\n",a,b,c);
23
24     return EXIT_SUCCESS;
25 }
26
27 /* definitions des fonctions utilisateurs */

```

2. Faire la trace du programme en considérant que l'utilisateur saisit au clavier : 1 puis "entrée", 12.2 puis "entrée" et 'c' puis "entrée" .