

## TP N°2

# Les instructions de contrôle

### 1. Question de mention au bac (utilisation du *if*)

Ecrivez un programme, dans un fichier "Exercice1.c", qui lit au clavier une moyenne et qui affiche la mention associée :

*Insuffisant si  $m < 10$*   
*Passable si  $m \geq 10$  et  $m < 12$*   
*Assez bien si  $m \geq 12$  et  $m < 14$*   
*Bien si  $m \geq 14$  et  $m < 16$*   
*Très bien si  $m \geq 16$*

Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```
./TP2EX1
Question de mention au bac :
-----
Entrez la moyenne > 12.4
La mention obtenue est : Assez bien
```

### 2. Chiffres romains (utilisation du *switch*)

Ecrivez un programme, dans un fichier "Exercice2.c", qui demande à l'utilisateur de saisir un chiffre romain et qui affiche le caractère correspondant. La liste des chiffres romains est :

- I = 1
- V = 5
- X = 10
- L = 50
- C = 100
- D = 500
- M = 1000

Attention on prendra soin de vérifier que la saisie du chiffre est correcte. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```
./TP2EX1
Chiffres romains :
-----
Entrez un chiffre romain > F
F n'est pas un chiffre romain.
Entrez un chiffre romain > X
La valeur de X est : 10
```

### 3. Je veux un nombre positif (utilisation du *while*, *do...while*)

Ecrivez un programme, dans un fichier "Exercice3.c", qui demande à l'utilisateur de saisir un entier au clavier tant que celui-ci est négatif. On prendra soin de vérifier que le choix du menu est correct. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```
./TP2EX3
Je veux nombre positif :
-----
Vous voulez utiliser un :
    1 - while
    2 - do while
Choix > 1
Entrez un nombre positif ou nul > -5
L'entier saisi est négatif
Entrez un nombre positif ou nul > -3
L'entier saisi est négatif
Entrez un nombre positif ou nul > 12
Merci
```

### 4. Les lettres de l'alphabet (utilisation du *for*)

Ecrivez un programme, dans un fichier "Exercice4.c", qui affiche les lettres de l'alphabet selon l'ordre croissant ou décroissant et en minuscules ou majuscules. On prendra soin de vérifier que le choix du menu est correct. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```
./TP2EX4
Les lettres de l'alphabet :
-----
Vous voulez un :
    1 - ordre croissant et minuscules
    2 - ordre croissant et majuscules
    3 - ordre decroissant et minuscules
    4 - ordre decroissant et majuscules
Choix > 3
Les lettres sont : z y x w v u t s r q p o n m l k j i h g f e d c b a
```

### 5. Calcul de la moyenne

Ecrivez un programme, dans un fichier "Exercice5.c", qui lit au clavier une suite x0, x1, x2, ... d'entiers positifs ou nuls et qui en affiche la moyenne. La frappe d'un nombre négatif indique la fin de la série. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```
./TP2EX5
Calcul de la moyenne :
-----
Entrez un nombre positif ou nul (negatif pour finir) > 5
Entrez un nombre positif ou nul (negatif pour finir) > 3
Entrez un nombre positif ou nul (negatif pour finir) > 0
Entrez un nombre positif ou nul (negatif pour finir) > 2
Entrez un nombre positif ou nul (negatif pour finir) > -1
Moyenne : 2.5
```

## 6. Triangle isocèle

Ecrivez un programme, dans un fichier "Exercice6.c", qui affiche un triangle isocèle formé d'étoiles. La hauteur du triangle (le nombre de lignes) sera entrée au clavier. On s'arrangera pour que la dernière ligne du triangle soit alignée sur le bord gauche de l'écran. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```
./TP2EX6
Triangle isocèle :
-----
Entrez la hauteur du triangle > 5
  *
 ***
*****
*****
*****
*****
```

## 7. Date du lendemain

Ecrivez un programme, dans un fichier "Exercice7.c", qui lit la date d'un jour, exprimée sous la forme de trois nombres (jour, mois et année) et qui calcul et affiche la date du lendemain. On supposera que la date donnée est correcte. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```
./TP2EX7
Date du lendemain :
-----
Entrez une date (j m a) > 10 9 2012
La date du lendemain est : 11 9 2012
```

*Rappel : Une année bissextile est une année qui vaut 0 modulo 4.*

## 8. Tri de trois entiers

Ecrivez un programme, dans un fichier "Exercice8.c", qui tri, de façon croissante, trois entiers entrés au clavier. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```
./TP2EX8
Tri de trois entiers :
-----
Entrez les trois entiers > 5 2 3
Les entiers tries sont : 2 3 5
```

## 9. Min/Max d'une suite de notes

Ecrivez un programme, dans un fichier "Exercice9.c", qui demande à l'utilisateur de rentrer une suite de note, entre 0 et 20, au clavier et qui trouve le maximum et le minimum. La saisie s'arrête lors d'une saisie négative. On pourra aussi afficher le nombre de fois où ces notes ont été attribuées. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```

./TP2EX9
Min/Max d'une suite de notes :
-----
Entrez une note entre 0 et 20 (négatif pour finir) > 0
Entrez une note entre 0 et 20 (négatif pour finir) > 22
Mauvaise saisie.
Entrez une note entre 0 et 20 (négatif pour finir) > 8
Entrez une note entre 0 et 20 (négatif pour finir) > 10
Entrez une note entre 0 et 20 (négatif pour finir) > 10
Entrez une note entre 0 et 20 (négatif pour finir) > -1
Note minimum : 0 donné 1 fois
Note maximum : 10 donné 2 fois

```

## 10. Racines carrés

Ecrivez un programme, dans un fichier "Exercice10.c", qui calcul la racine carré des nombres entiers positifs entrés par l'utilisateur au clavier. Le programme se termine à la saisie d'un nombre négatif. Le programme doit avoir le fonctionnement suivant (en rouge les saisies de l'utilisateur) :

```

./TP2EX10
Racines carres :
-----
Entrez un entier positif (négatif pour quitter) > 4
La racine carre de 4 est 2
Entrez un entier positif (négatif pour quitter) > 2
La racine carre de 2 est 1.414214
Entrez un entier positif (négatif pour quitter) > -1

```

*Rappel : on utilisera la fonction sqrt de la bibliothèque math.h (#include <math.h>) pour le calcul de la racine carré. Il faudra rajouter -lm à la fin de la ligne de compilation pour rajouter la librairie mathématique.*

## 11. Table de multiplication

Ecrivez un programme, dans un fichier "Exercice11.c", qui affiche la table de multiplication des nombres de 1 à 10. Le programme doit avoir le fonctionnement suivant :

```

./TP2EX11
Table de multiplication :
-----
| 1  2  3  4  5  6  7  8  9 10
1 | 1  2  3  4  5  6  7  8  9 10
2 | 2  4  6  8 10 12 14 16 18 20
3 | 3  6  9 12 15 18 21 24 27 30
4 | 4  8 12 16 20 24 28 32 36 40
5 | 5 10 15 20 25 30 35 40 45 50
6 | 6 12 18 24 30 36 42 48 54 60
7 | 7 14 20 28 35 42 49 56 63 70
8 | 8 16 24 32 40 48 56 64 72 80
9 | 9 18 27 36 45 54 63 72 81 90
10 | 10 20 30 40 50 60 70 80 90 100

```

## 12. Combinaisons de centimes

Ecrivez un programme, dans un fichier "Exercice12.c", qui permet d'afficher toutes les possibilités d'obtenir 1 euros avec des pièces de 1, 2, 5, 10, 20 et 50 centimes. On affichera combien de possibilités ont été trouvées. Le programme doit avoir le fonctionnement suivant :

```
./TP2EX12
Combinaisons de centimes :
-----
1 euro = 2 x 50c
1 euro = 5 x 20c
1 euro = 1 x 50c  2 x 20c  1 x 10c
1 euro = 4 x 20c  2 x 10c
1 euro = 1 x 50c  1 x 20c  3 x 10c
...
1 euro = 3 x 2c  94 x 1c
1 euro = 1 x 5c  95 x 1c
1 euro = 2 x 2c  96 x 1c
1 euro = 1 x 2c  98 x 1c
1 euro = 100 x 1c
Il y a au total 4562 combinaisons.
```