
Travaux pratiques 4 : la structure de contrôle “for”

L’objectif de ce TP est de vous familiariser avec les notions d’itération et de boucles imbriquées.

Correction. Il y a deux exos type qu’il faut corriger au tableau. Le dernier exo du TD4 (Affichage de n fois “Bonjour”) peut être fait en intro si pas fini pendant le TD.

Vous allez mettre tous vos programmes écrits dans ce TP dans le répertoire TP4.

1. À partir du début de votre arborescence, créez le répertoire TP4 : `mkdir TP4`
2. Allez dans ce répertoire pour y mettre des fichiers : `cd TP4`

L’étape suivante est à répéter pour chaque nouveau programme (exo1, exo2 etc..) :

3. Créez un nouveau fichier source pour le langage C ou une nouvelle copie d’un programme existant.

Création `kwrite exo1.c` & (vous pouvez utiliser `emacs` au lieu de `kwrite`)

Copie Il est plus rapide de repartir d’une copie de votre programme `bonjour.c` du TP2 pour éviter de retaper tout le squelette. Dans le terminal :

```
cp ../TP2/bonjour.c exo1.c
kwrite exo1.c &
```

Vous pouvez-aussi ouvrir `bonjour.c` et utiliser la fonction *Enregistrer sous...* de votre éditeur mais attention à enregistrer la nouvelle copie dans le bon répertoire.

Vous pouvez utiliser à tout moment la commande `ls` (list directory) pour voir la liste des fichiers d’un répertoire.

Les trois étapes suivantes seront à répéter autant de fois que nécessaire pour la mise au point de chaque programme (apprenez à utiliser les raccourcis clavier).

4. Après avoir fini d’écrire votre programme, enregistrez le.
5. Créez un programme exécutable à partir de votre fichier source :
`gcc -Wall exo1.c -o exo1.exe`
6. Quand l’étape précédente a réussi, exécutez le programme pour vérifier qu’il fonctionne :
`exo1.exe` (ou `./exo1.exe`).

1 Affichage de figures géométriques

Les exercices suivants utilisent le caractère * (étoile) pour dessiner des figures géométriques simples, appelées figures d’étoiles.

1.1 Exercice type : affichage d'un rectangle d'étoiles

Écrire un programme qui, étant données deux variables, `longueur` et `largeur`, initialisées à des valeurs strictement positives quelconques, affiche un rectangle d'étoiles ayant pour longueur `longueur` étoiles et largeur `largeur` étoiles. Deux exemples d'exécution, avec deux initialisations différentes, sont les suivants :

Affichage d'un rectangle d'étoiles de longueur 10 et largeur 5.

```
*****
*****
*****
*****
*****
```

Affichage d'un rectangle d'étoiles de longueur 6 et largeur 3.

```
*****
*****
*****
```

Correction. Durée 3/4 d'heure ?

Les algos sont à faire (les extraire du code).

Vous pouvez dans un premier temps supprimer la boucle la plus imbriquée en leur demandant d'afficher un rectangle de longueur exactement "*****".

```
/* declaration de fonctionnalites supplementaires */
#include <stdlib.h> /* EXIT_SUCCESS */
#include <stdio.h> /* printf */

/* declaration constantes et types utilisateurs */

/* declaration de fonctions utilisateurs */

/* fonction principale */
int main()
{
    /* declaration et initialisation variables */
    int largeur = 3; /* largeur du rectangle en nb d'étoiles */
    int longueur = 6; /* longueur du rectangle en nb d'étoiles */
    int i; /* var. de boucle */
    int j; /* var. de boucle */

    printf("Affichage d'un rectangle d'étoiles de longueur %d et largeur %d.\n",longueur,largeur);

    for(i = 0;i < largeur;i = i + 1) /* chaque ligne d'étoiles */
    {
        /* affiche longueur étoiles */
        for(j = 0;j < longueur;j = j + 1) /* chaque colonne d'étoiles */
        {
            /* affiche une étoile */
```

```

        printf("*");
    }
    /* j >= longueur */

    /* passe a la ligne suivante */
    printf("\n");
}
/* i >= largeur */

return EXIT_SUCCESS;
}

/* implantation de fonctions utilisateurs */

```

1.2 Exercice type : affichage d'un demi-carré d'étoiles

Écrire un programme qui affiche, étant donnée la variable, *cote*, initialisée à une valeur quelconque, un demi-carré d'étoiles (triangle rectangle isocèle) ayant pour longueur de côté *cote* étoiles. Deux exemples d'exécution, avec deux initialisations différentes, sont les suivants :

Affichage d'un demi-carre d'etoiles de cote 6.

```

*
**
***
****
*****
*****

```

Affichage d'un demi-carre d'etoiles de cote 2.

```

*
**

```

Correction. Durée 3/4 d'heure?

Les algos sont à faire (les extraire du code).

```

/* declaration de fonctionnalites supplementaires */
#include <stdlib.h> /* EXIT_SUCCESS */
#include <stdio.h> /* printf */

/* declaration constantes et types utilisateurs */

/* declaration de fonctions utilisateurs */

/* fonction principale */
int main()
{
    /* declaration et initialisation variables */
    int cote = 2; /* cote du demi-carré en nb d'etoiles */

```

```

int i; /* var. de boucle */
int j; /* var. de boucle */

printf("Affichage d'un demi-carre d'etoiles de cote %d.\n",cote);

for(i = 1;i <= cote;i = i + 1) /* chaque numero de ligne d'etoiles */
{
    /* affiche autant d'etoiles que le numero de ligne */
    for(j = 0;j < i;j = j + 1) /* chaque colonne d'etoiles */
    {
        /* affiche une etoile */
        printf("*");
    }
    /* j >= i */

    /* passe a la ligne suivante */
    printf("\n");
}
/* i > cote */

return EXIT_SUCCESS;
}

/* implantation de fonctions utilisateurs */

```