

Travaux dirigés 9 : écriture et appel de fonctions et procédures (2)

1 Trace de fonctions

1. Faire la trace du programme suivant.

```
1  /* Declaration de fonctionnalites supplementaires */
2  #include <stdlib.h> /* EXIT_SUCCESS */
3  #include <stdio.h> /* printf() */
4
5  /* Declarations constantes et types utilisateurs */
6
7  /* Declarations de fonctions utilisateurs */
8  void permute_valeurs(int a,int b);
9
10 /* Fonction principale */
11 int main()
12 {
13     /* Declaration et initialisation des variables */
14     int x = 1;
15     int y = 2;
16
17     permute_valeurs(x,y);
18     printf("x = %d et y = %d\n",x,y);
19
20     /* Valeur fonction */
21     return EXIT_SUCCESS;
22 }
23
24 /* Definitions de fonctions utilisateurs */
25 void permute_valeurs(int a,int b)
26 {
27     int aux; /*var aux pour permutation */
28
29     aux = a;
30     a = b;
31     b = aux;
32 }
```

2 Le menu avec fonctions et procédures

Dans cet exercice vous complétez le menu écrit en TP. Vous travaillerez sur trois parties du programme :

- les déclarations du début du programme (fonctionnalités, constantes, fonctions),
- les définitions de fonctions,
- la fonction principale (main).

```
***** MENU *****
*
*  1) Tester si un nombre est premier
*  2) Calcul de factorielle
*  3) Deviner un nombre
*  4) Motif d'etoiles
*
*  0) QUITTER
*
***** votre choix :

```

2. Déclarer et définir une procédure qui affichera le menu.
3. Déclarer et définir une fonction `choix_utilisateur`, sans paramètres qui renverra une valeur entière saisie par l'utilisateur.

4. Déclarer et définir une fonction `executer_menu` qui :
 - affichera le menu à l'utilisateur et réalisera la saisie de son choix ;
 - lorsque ce choix est 1, appellera une fonction non encore définie `menu_premier` ;
 - puis, lorsque ce choix est différent de 0, renverra `TRUE` et lorsque ce choix est égal à zéro renverra `FALSE`.
5. Déclarer et définir une procédure `menu_premier` qui traitera le choix 1, et qui fera appel à la fonction `est_premier` du TP (dont vous rappellerez la définition et la déclaration) et à la fonction `choix_utilisateur` pour le choix de l'entier.
6. Écrire le `main` de telle sorte qu'il fasse appel à la fonction `executer_menu` tant que celle-ci renvoie `TRUE`.
7. Faire la trace de votre programme dans le cas où l'utilisateur saisit 1 puis 3 puis 0.
8. Modifier votre fonction `choix_utilisateur` de telle sorte que :
 - elle prenne en argument deux paramètres entiers a et b ;
 - si l'utilisateur saisit un nombre $n \in [a, b]$ la fonction retourne n sans générer d'affichage ;
 - si l'utilisateur saisit un nombre $n \notin [a, b]$ l'intervalle de saisie soit affiché à l'utilisateur et la saisie redemandée, jusqu'à cinq fois.

On se donne la procédure suivante :

```
void affichage_motif(int cote)
{
    int ligne; /* numero de ligne, de bas en haut */
    int colonne; /* numero de colonne, de gauche a droite */
    for (ligne = cote - 1; ligne >= 0; ligne = ligne - 1) /* pour chaque ligne */
    {
        for (colonne = 0; colonne < cote; colonne = colonne + 1) /* pour chaque colonne */
        {
            if (motif(colonne, ligne)) /* le point appartient au motif */
            {
                printf("* ");
            }
            else /* le point n'appartient pas au motif */
            {
                printf("  ");
            }
        }
        printf("\n"); /* ligne suivante */
    }
}
```

9. Définir la fonction `motif` de telle sorte qu'un appel à `affichage_motif(5)` affiche :

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

10. Écrire les fonctions nécessaires au traitement du choix 3 du menu sur le modèle du traitement du choix 1.

Programme à tester et compléter en TP.