Auteur : Mohamed Messabihi
Matière : Programmation et structures de données
Mohamed.Messabihi@gmail.com

Université Abou Bakr Belkaïd - Tlemcen
Faculté des sciences
1ère L.M.D. MI

5 TD : Les pointeurs

5.1 Compléter le programme

Compléter le programme ci-dessous en se basant sur les idées présentes dans les messages des printf.

```c
#include <stdio.h>
int main(void){
    int n=4,  *ptr1 = &n;;
    double x = 6.38, *ptr2 = &x;
    printf("********** Operateurs (&) vs. (*) **********\n");
    printf("\n*******Les valeurs avec des variables :*****\n");
    printf( " n = ......\n", ......);
    printf( " x = ......\n", ......);
    printf("\n*****Les adresses avec des variables :******\n");
    printf( " Adresse de n = ......\n", ...n...);
    printf( " Adresse de x = ......\n", ...x...);
    printf("\n******Les adresses avec des pointeurs :******\n");
    printf( " L'adresse de n = ......\n", ...ptr1...);
    printf( " L'adresse de x = ......\n", ...ptr2...);
    printf("\n******Les valeurs avec des pointeurs :******\n");
    printf( " La valeur a l'adresse de n= ......\n", ...ptr1...);
    printf( " La valeur a l'adresse de x= ......\n", ...ptr2...);
}
```

5.2 Ce n'est qu'une histoire d'adresses

Montrez l'historique d'exécution détaillé ainsi que l'affichage des deux programmes suivants :

1. Programe **abcxy**

```c
#include <stdio.h>
int main(void){
    int x=1, y=2;
    int *a = &x, *b = &y, *c = a;
    printf("*a = %d, *b = %d, *c = %d\n", *a, *b, *c);
    a = b;
    printf("*a = %d, *b = %d, *c = %d\n", *a, *b, *c);
    *a = 3;
    printf("*a = %d, *b = %d, *c = %d\n", *a, *b, *c);
    *c = 4;
    printf("x = %d, y = %d\n", x, y);
    *b = *c;
    if(b == c)
        *a = 5;
    else
        *a = 6;
    printf("*a = %d\n", *a);
}
```

2. Programme **Toto & Loulou**

```c
#include <stdio.h>
void toto (int *a, int *b){
    int c;
    c = *a ;
    *a = *b ;
    *b = c ;
    printf ("Dans toto : *a=%d, *b=%d\n", *a, *b);
}
void loulou (int *a, int *b){
    int *c;
    c = a ;
    a = b ;
    b = c ;
    printf ("Dans loulou : *a=%d, *b=%d\n", *a, *b);
}
void main(){
    int i=3, j=6, n=4, m=8 ;
    toto(&i, &j) ;
    printf ("Apres Toto : i=%d, j=%d\n", i, j);
    loulou (&n, &m) ;
    printf ("Apres Loulou ; n=%d, m=%d\n", n, m);
}
```

En déduire le rôle de la fonction toto() et celui de la fonction loulou().

5.3 Trier

Écrire une fonction Trier qui ordonne les valeurs de trois variables entières dans l'ordre croissant. Cette fonction aura trois paramètres et renverra 0 si les variables étaient déjà dans l'ordre (et donc la fonction n'aura pas modifié l'ordre), ou 1 si elle a vraiment ordonné les variables.

Écrire une fonction main pour tester la fonction trier qui doit ordonner les valeurs des variables reçues depuis la fonction main.

Exemple. si dans la fonction main on a X=3, Y=1, Z=2, après l'appel de la fonction Trier on obtient : X=1, Y=2, Z=3.

5.4 Simplification de fraction

Écrire une fonction SimplifierFraction qui simplifie une fraction de nombres entiers en modifiant les valeurs du numérateur et du dénominateur (cette fonction aura donc deux paramètres). Utiliser pour cela la fonction **PGCD** de l'exercice 4.3 du TD précédent.

Écrire une fonction main pour tester la fonction SimplifierFraction qui doit modifier les valeurs des variables reçues depuis la fonction main.

Exemple. si dans la fonction main on a n=36 et d=16, après l'appel de la fonction SimplifierFraction on obtient : n=9 et d=4.