

Devoir Maison – Corrigé

Enoncé

A l'aide des concepts vu en cours, écrire un programme qui demande à l'utilisateur 3 nombres et les affiche en retour triés par ordre croissant. Idem pour 4 nombres. Par exemple, si l'utilisateur fournit au programme les nombres « 2 », « -1 », « 3 », « 10 », le programme doit les afficher dans l'ordre « -1 2 3 10 ».

Le devoir devra être rendu sur papier et le code commenté et expliqué (comment il marche et comment il trie les nombres). Il doit contenir également une trace d'exécution du programme proposé en supposant que l'utilisateur a fourni la séquence « 2 », « -1 », « 3 », « 10 ».

Le devoir est à faire en binôme et doit être rendu au chargé de TD référant à la semaine 48 (du 24 au 28 novembre), pendant la séance de TD appropriée.

Correction

Puisque les étudiants n'ont pas encore vu les vecteurs, ils sont donc obligés à utiliser une multitude des tests *if* avec les différentes possibilités. Grosso modo, les étudiants auront besoin d'au moins 3 *if* pour trier 3 numéros, et au moins 5 pour trier 4 numéros.

Aucun usage de vecteurs ne doit être toléré ! Les étudiants ont été avertis du fait qu'ils doivent utiliser uniquement les concepts vu en cours. Ceci a pour but de les obliger à travailler par eux-mêmes et d'éviter les copies sur Internet, puisque la majorité des solutions proposées sur Internet utilisent des vecteurs. Par ailleurs, les tests *if* ont été vus en cours, cet exercice valorise donc les étudiants qui viennent en cours et qui préparent leur TDs.

En ce qui concerne les barèmes, je vous propose la notation suivante (évidement, vous êtes libres pour choisir celle qui vous convient au mieux) :

- 5pts pour la trace d'exécution
- 5pts pour les commentaires / explications sur le code (voir s'ils ont bien compris ce qu'ils proposent)
- 10pts pour le code
 - -1pt pour if mal défini
 - -1pt pour variable non-déclarée
 - -1pt pour printf/scanf mal rédigé

Cet exercice est un exercice du poly de *M. Nicolas Trotignon*, utilisé l'année précédente. Celui-ci nous propose plusieurs solutions décrites dans le code ci-dessous :

```
#include<stdio.h>

main()
{
    int a, b, c, d;
    int aa, bb, cc, dd;
    int t;

    printf("Entrez un entier a : ");
    scanf("%i", &a);
    printf("Entrez un entier b : ");
    scanf("%i", &b);
    printf("Entrez un entier c : ");
    scanf("%i", &c);
    printf("Entrez un entier d : ");
    scanf("%i", &d);

    aa=a;
    bb=b;
    cc=c;
    dd=d; //On sauvegarde les valeurs pour faire plusieurs tests

    // Premiere methode : lourd
    if (a <= b && b <= c) printf("%i, %i, %i\n", a, b, c);
    if (a <= c && c <= b) printf("%i, %i, %i\n", a, c, b);
    if (b <= a && a <= c) printf("%i, %i, %i\n", b, a, c);
    if (b <= c && c <= a) printf("%i, %i, %i\n", b, c, a);
    if (c <= a && a <= b) printf("%i, %i, %i\n", c, a, b);
    if (c <= b && b <= a) printf("%i, %i, %i\n", c, b, a);
    a=aa;
    b=bb;
    c=cc;
    d=dd; //On remet `a jour les valeurs

    //Une solution mixte
    if (a>b) {
        t=a;    //Maintenant, a<=b.
        a=b;
        b=t;
    }
    if (c<=a) printf("%i, %i, %i\n", c, a, b);
    else
        if (c<=b) printf("%i, %i, %i\n", a, c, b);
        else
            printf("%i, %i, %i\n", a, b, c);
    a=aa;
    b=bb;
    c=cc;
    d=dd; //On remet `a jour les valeurs
```

```

//Une solution qui trie vraiment
if (a>b) {
    t=a;    //Maintenant, a<=b. restent 3 cas
    a=b;
    b=t;
}
if (a>c) {
    t=a;    //Maintenant, a<=c. restent 2 cas
    a=c;
    c=t;
}
if (b>c) {
    t=b;    //Maintenant, b<=c.
    b=c;
    c=t;
}
//Maintenant a<=b<=c
printf("%i, %i, %i\n", a, b, c);
a=aa;
b=bb;
c=cc;
d=dd;
//On remet `a jour les valeurs

//Une solution pour 4 nombres avec seulement 5 if
if (a>b) {
    t=a;    //Maintenant, a<=b. restent 12 cas
    a=b;
    b=t;
}
if (c>d) {
    t=c;    //Maintenant, c<=d. restent 6 cas
    c=d;
    d=t;
}
if (b>d) {
    t=b;    //Maintenant, b<=d. restent 3 cas
    b=d;
    d=t;
}
if (a>c) {
    t=a;    //Maintenant, a<=c. restent 2 cas
    a=c;
    c=t;
}
if (b>c) {
    t=b;    //Maintenant, b<=c.
    b=c;
    c=t;
}
//Maintenant a<=b<=c<=d
printf("%i, %i, %i, %i\n", a, b, c, d);
//Peut-on ecrire une solution aussi simple pour 5 nombres ? Mystere...
}

```