

# TP2 de LE1 IPA : Exemple de compte-rendu

Auteur : Armand TOGUYENI

Date : 09/11/2020

## 1. Tests fonctionnels de méthodes de tri

Il s'agit de faire des tests avec des tableaux de petites tailles afin de pouvoir vérifier manuellement que le tri est bien effectué.

- a. Vue d'ensemble

*Entrer la taille du tableau à generer :* 10

**1 - Generation d'un tableau avec un ordre aleatoire**

**2 - Generation d'un tableau dans l'ordre croissant**

**3 - Generation d'un tableau dans l'ordre decroissant**

**Choix de la methode de generation :** 1

**Voulez-vous afficher le tableau (O/N):** O

684.24 596.60 504.76 570.50 305.78 122.57 843.88 866.67 621.39 294.65 252.92

**a - Tri selection**

**b - Tri echange**

**c - Tri insertion**

**Choix methode de tri :**

a

**Tri selection**

**Le tri selection d'un tableau de 10 elements a pris 1 us !!!**

**Voulez-vous afficher le tableau (O/N):** O

122.57 294.65 305.78 504.76 570.50 596.60 621.39 684.24 843.88 866.67 252.92 **Voulez-vous poursuivre avec une autre methode de tri et le meme tableau (O/N) :** O

**a - Tri selection**

**b - Tri echange**

**c - Tri insertion**

**Choix methode de tri :**

b

**Tri echange**

**Le tri echange d'un tableau de 10 elements a pris 1 us !!!**

**Voulez-vous afficher le tableau (O/N): O**

**122.57 252.92 294.00 305.00 504.00 570.00 596.00 621.00 684.00 843.00 866.00 Voulez-vous poursuivre avec une autre methode de tri et le meme tableau (O/N) :O**

**a - Tri selection**

**b - Tri echange**

**c - Tri insertion**

**Choix methode de tri :**

**C**

**Tri insertion**

**Le tri insertion d'un tableau de 10 elements a pris 0 us !!!**

**Voulez-vous afficher le tableau (O/N): O**

**122.57 294.65 305.78 504.76 570.50 596.60 621.39 684.24 843.88 866.67 252.92 Voulez-vous poursuivre avec une autre methode de tri et le meme tableau (O/N) :N**

**Voulez-vous poursuivre l'application (O/N) :N**

**FIN de l'application de Tris !!!**

**1. Tri par selection**

**Entrer la taille du tableau a generer : 10**

**1 - Generation d'un tableau avec un ordre aleatoire**

**2 - Generation d'un tableau dans l'ordre croissant**

**3 - Generation d'un tableau dans l'ordre decroissant**

**Choix de la methode de generation : 1**

**Voulez-vous afficher le tableau (O/N): O**

**684.24 596.60 504.76 570.50 305.78 122.57 843.88 866.67 621.39 294.65 252.92**

**a - Tri selection**

**b - Tri echange**

**c - Tri insertion**

**Choix methode de tri :**

**a**

**Tri selection**

**Le tri selection d'un tableau de 10 elements a pris 1 us !!!**

**Voulez-vous afficher le tableau (O/N): O**

**122.57 294.65 305.78 504.76 570.50 596.60 621.39 684.24 843.88 866.67 252.92 Voulez-vous poursuivre avec une autre methode de tri et le meme tableau (O/N) :O**

Commentaire : On la valeur 252.92 est mal positionnée car se trouvant après 866.67 qui est la valeur la plus petite. Cela peut être dû à 3 problèmes :

Hypothèse 1 : L'implémentation d'une mauvaise méthode de tri

Hypothèse 2 : Le tableau généré est trop grand par rapport à la dimension saisie.

Hypothèse 3 : La méthode de tri ne s'applique pas à tout le tableau.

Quand on compte le nombre d'éléments générés du tableau, on s'aperçoit qu'il y en a 11 et non 10.

**Entrer la taille du tableau a generer : 10**

**1 - Generation d'un tableau avec un ordre aleatoire**

**2 - Generation d'un tableau dans l'ordre croissant**

**3 - Generation d'un tableau dans l'ordre decroissant**

**Choix de la methode de generation : 1**

**Voulez-vous afficher le tableau (O/N): O**

**684.24 596.60 504.76 570.50 305.78 122.57 843.88 866.67 621.39 294.65 252.92**

L'affichage du tableau généré montre que 252.92 est la dernière valeur. Donc, le test semble montrer que c'est l'hypothèse 2 qui serait la bonne. Il faut vérifier le code.

```
//Affichage du tableau genere s'il n'est pas trop grand
printf("Voulez-vous afficher le tableau (O/N): ");
rep=getchar();
getchar();
if ((tolower(rep)=='o') && (nb<=MAXAFF))
{
    for(i=0; i<nb; i++)
    {
        if (!(i % MAXL)) printf("\n"); //On passe à la ligne suivante chaque fois que l'on
        printf("%.2lf \t",T1[i]);
    }
} else
    if (nb>MAXAFF) printf("Affichage impossible car le tableau a une taille supérieure à %d\n");
    printf("\n"); //Passage de l'affichage à la ligne suivante
```

Figure 1. Code de l'affichage dans tp2\_tri\_squelette.c

On voit sur le code de l'affichage (Figure 1) que les composantes du tableau sont affichées de 0 à nb. Comme dans le test nb vaut 10, cela signifie qu'il y a une erreur au niveau de l'affichage. Il faudrait que la condition de poursuite de la boucle « for » soit « **i<=nb** ».

## b - Tri par échange

### Tri echange

Le tri echange d'un tableau de 10 elements a pris 1 us !!!

Voulez-vous afficher le tableau (O/N): O

122.57 252.92 294.00 305.00 504.00 570.00 596.00 621.00 684.00 843.00 866.00 Voulez-vous poursuivre avec une autre méthode de tri et le même tableau (O/N) :O

Figure 2. Affichage du tableau trié par la méthode du tri échange (tri à bulles)

Sur la Figure 2, on voit que le tableau est trié, mais de la 3<sup>ème</sup> valeur à la dernière valeur, tous les nombres réels ont été arrondis.

Quand on regarde le code, on voit qu'une variable appelée « aux » est utilisée pour faire les échanges.

```
//Déclaration des variables
double T1[MAXCOMP], T2[MAXCOMP] ; //Tableaux de MAXCOMP d'entiers
int i; //Compteur de boucle
int nb; //dimension du tableau
int duree;
struct timeval debut, fin ;
char rep; //Saisie réponse du user
int choix; //Saisie réponse user pour le choix de la méthode de génération du tableau
int aux;
```

Figure 3. Déclaration des variables du programme

Sur la Figure 3, on constate que cette variable est de type « **int** », alors que les données du tableau sont des « **double** ». On voit donc que le mauvais type « **int** » de la variable est responsable de l'arrondi des valeurs du tableau résultat affiché.

## 2. Tests de performances

Les tests vont être faits avec différentes valeurs de manière à corrélérer l'efficacité des méthodes de tri par rapport à la taille du tableau.

Tableau généré de manière aléatoire						
Quantité d'éléments à trier	100	1000	5000	10000	20000	300000
Méthode de tri						
Tri sélection						
Tri à bulles (Tri échanges)						
Tri insertion						

Figure 4. Efficacité des méthodes de tri par rapport à un tableau au désordre aléatoire.

## 3. Conclusion

Dans un compte-rendu on ne se contente pas d'afficher les résultats. Il faut les commenter.

Il faut également savoir prendre du recul par rapport aux résultats obtenus.