

Rapport

Le but de ce programme est de faire un tirage d'une matrice. Le programme a été faite en C++ avec l'environnement DEV-C++. Le programme commence par déclarer toutes les variables nécessaires pour faire lire les commandes. Le conteneur que j'utilise pour le programme est le conteneur Vector. La raison pourquoi j'utilise le conteneur Vector est parce que le Vector est simple a utilisé et n'a pas besoin de grandeur spécifique. Le Vector peut grandir ou diminuer en grandeur par les fonctions `push_back` et `pop_back`. Cela facilite pour la taille de la matrice parce qu'il y a deux moyens de remplir la matrice et on ne connaît pas la taille à chaque fois pour utiliser un Array. Lorsqu'on lit les commandes, les signe pour chaque commande est activer par le Boolean. Le Boolean va indiquer au programme si tell commandes a été écrite par l'user. Avec la commande -a, on regarde si la commande -r a été écrit. Si -r a été mis on fait un `srand (time ())` qui permet d'avoir des valeur aléatoire random et non le même jeu à chaque fois. Sans le -r on continue avec le programme. Le programme a 3 valeurs aléatoire, 1 pour la colonne, 1 pour la rangée, 1 pour remplir la matrice avec les valeurs. La colonne et la rangée est faite avec $\%10 + 1$ parce que on veut des valeurs entre 1 et 10. Les valeur pour remplie est faite avec $\%10$ pour avoir des valeurs entre 0 et 9. On crée un Vector temporaire pour chaque range et on le `push_back` dans notre matrice. Pour la commande -o on lit le fichier dans lequel on veut sauvegarder et on écrit la matrice avant et après le triage. Pour la commande -i on lit le fichier duquel on lit pour remplir la matrice. On crée un Vector temporaire pour lire les valeurs en rangée et on le `push_back` dans notre matrice. Lorsqu'on fait le triage default on fait un triage part rangée. On compare toutes les valeurs dans la position 0 de la colonne de chaque rangée. Si la valeur est plus petite qu'un autre on effectue un switch. Le triage est fait par la fonction `sort` qui est donne dans le STL. S'il y a la commande -c pour le triage par colonne on utilise un algorithme et non une fonction. L'algorithme effectue le triage et lorsqu'il y a une valeur plus petite que l'autre dans la rangée il effectue un changement de toutes les valeurs de la même colonne. Les difficultés que j'ai durant ce programme est comment lire un fichier, écrire dans un fichier et faire le triage en colonne. Pour lire et écrire le fichier j'ai utilisé des tutoriels en ligne qui explique comment la faire. Le programme utilise le `#include <fstream>` pour tous ce qui est relire avec le fichier. Pour le triage en colonne, j'ai utilisé des exemples pour me donner une idée. Le tirage par rangée faite manuellement ma beaucoup aider. Si j'avais eu plus de temps à améliorer le programme, je pouvais faire le programme pour qu'il accepte toute sort de type avec le type Template.

Voici les résultats des commandes.

-h

```
C:\Users\Shurid>cd Desktop\TP1-1169

C:\Users\Shurid\Desktop\TP1-1169>g++ tp1int.cpp -o main

C:\Users\Shurid\Desktop\TP1-1169>main -h
      La Matrice
Le but de ce programme est de trier une matrice sur les rangees ou les colonnes
avec des donnee aleatoire ou du user
-o output: le fichier qui suit la commande va contenir les resultat
-i input: le fichier qui suit la commande va contneir les valeurs a utiliser
-a aleatoire: va generer des tailles et valeur aleatoires
-r random: va peremettre a la commande -a de generer de nouvelles valeur et non le meme
-c colonne: triage faite en colonnes
-h help: pour savoirs les different commandes

C:\Users\Shurid\Desktop\TP1-1169>
```

-a

```
C:\Users\Shurid\Desktop\TP1-1169>main -a
-a aleatoire.
The Matrix before sorting is:
4 0 9 4 8 8 2 4
5 5 1 7 1 1 5 2
The Matrix after sorting is:
4 0 9 4 8 8 2 4
5 5 1 7 1 1 5 2

C:\Users\Shurid\Desktop\TP1-1169>
```

-c (avec -a)

```
C:\Users\Shurid\Desktop\TP1-1169>main -c -a
-c tirage faite en colonne
-a aleatoire.
The Matrix before sorting is:
4 0 9 4 8 8 2 4
5 5 1 7 1 1 5 2
The Matrix after sorting is:
0 2 4 4 4 8 8 9
5 5 5 7 2 1 1 1

C:\Users\Shurid\Desktop\TP1-1169>
```

-r (avec -a)

```
C:\Users\Shurid\Desktop\TP1-1169>main -r -a
-r randomize
-a alleatoire.
The Matrix before sorting is:
9 6 6 9 0 3 9 8 7
1 5 7 2 1 4 8 8 9
1 5 2 5 9 6 1 4 0
8 6 3 7 4 7 9 7 7
The Matrix after sorting is:
1 5 7 2 1 4 8 8 9
1 5 2 5 9 6 1 4 0
8 6 3 7 4 7 9 7 7
9 6 6 9 0 3 9 8 7
```

-i (avec nom du fichier)

```
C:\Users\Shurid\Desktop\TP1-1169>main -i in1.txt
-i input. le fichier est in1.txt
The Matrix before sorting is:
4 2 2
3 3 6
3 3 1
The Matrix after sorting is:
3 3 6
3 3 1
4 2 2
C:\Users\Shurid\Desktop\TP1-1169>
```

-o (avec nom du fichier et -a)

```
C:\Users\Shurid\Desktop\TP1-1169>main -o out.txt -a
-o output. le fichier est out.txt
-a alleatoire.
```