

TD - TP 4

Objectifs : programmation en langage C

- Nombres pseudo-aléatoires
- Tableau, pointeur

Références

- **Chapter 4 : Pointers**
- **Chapter 5 : Vectors**

1. Ecrire un programme qui affiche à l'écran une suite pseudo-aléatoire de n nombres (n fixé constant ou lu au clavier) fournis par appel à la fonction `rand`. Utiliser le manuel en ligne pour connaître son mode d'emploi (commande `man 3 rand`).
2. Modifier ce programme pour mémoriser dans un tableau n nombres pseudo-aléatoires.
3. Ecrire une fonction de prototype : `void imptabT(int n, int t[])`; qui affiche à l'écran les n premiers éléments du tableau `t`. Tester cette fonction en ajoutant son appel au programme précédent.
4. Ecrire une fonction de prototype : `void imptabP(int n, int *t)`; qui affiche à l'écran les n valeurs situées en mémoire à partir de l'adresse contenue dans `t`. Utiliser des indices pour parcourir les valeurs. Peut-on reprendre la même programmation (la même écriture) que dans la fonction `imptabT` ? Que peut-on en conclure ? Tester cette fonction.
5. Ecrire une autre version de `imptabP`, en conservant le prototype précédent, sans utiliser d'indice. L'appel à cette nouvelle version dans la fonction `main` doit-il être changé ?
6. On souhaite afficher les 10 derniers nombres du tableau. Peut-on le faire à l'aide des fonctions `imptabT` et `imptabP` ? Si oui, comment ?
7. On souhaite avoir une idée de la répartition des valeurs fournies par la fonction `rand`. Pour cela, on considère une subdivision uniforme de l'intervalle dans lequel `rand` renvoie ses valeurs en `nbint` sous-intervalles et on effectue une série de N tirages en comptant le nombre de valeurs situées dans chaque sous-intervalle.
Ecrire un programme qui affiche ces nombres à l'écran en utilisant la fonction `imptabT` ou `imptabP`. Prendre par exemple `nbint = 10` et $N = 100\ 000$. Programmer cela le plus simplement possible et en minimisant la mémoire utilisée. On sera notamment confronté aux questions suivantes :
 - est-il nécessaire de mémoriser les valeurs fournies par `rand` ?
 - peut-on réaliser la programmation sans test (sans recourir à l'instruction `if` ou `switch`) ? Regrouper les déclarations de fonctions dans un fichier `decl.h` à inclure dans les fichiers concernés.
8. Etant donnée une suite de valeurs stockées dans un tableau, écrire une fonction qui calcule leur moyenne et une fonction qui calcule leur écart type. Utiliser ces fonctions dans le contexte de la question précédente pour estimer la qualité de la répartition des valeurs fournies par la fonction `rand`. Il faudra déclarer la fonction `sqrtrf` à l'aide du fichier `math.h` et faire l'édition de liens en ajoutant l'option `-lm` sur la ligne de commande.

Travail à rendre :

Rendre les fichiers sources relatifs aux deux dernières questions du TP en y joignant un bref mode d'emploi (création de l'exécutable, commentaire des résultats), à rédiger dans un autre fichier d'extension `.txt`. Remettre ces fichiers dans l'espace de dépôt Moodle "Compte rendu TP4".