
Le but du TD est de bien comprendre la ligne de commande pour rentrer les paramètres, les tableaux et les boucles For.

Exercice 1 : La ligne de commande

1.a] Écrire un programme qui additionne les nombres placés sur la ligne de commande, les affiche, affiche le nombre de paramètres et affiche la somme.

Exercice 2 : Les Tableaux

Initialiser un tableau avec des nombres entrés sur la ligne de commande (au plus 10 entiers). Les questions suivantes seront faites dans le même fichier.

2.a] Afficher le tableau.

2.b] Compter le nombre de valeurs supérieures à un seuil donné.

2.c] Afficher le tableau dans l'ordre inverse.

2.d] Inverser l'ordre des éléments du tableau et afficher le tableau.

2.e] Trouver le maximum et le minimum du tableau.

Exercice 3 : Tableau aléatoirement initialisé

Pour utiliser le générateur de nombre aléatoire, on utilisera les fonctions `srand` et `rand`. Quel est l'include nécessaire pour les utiliser ? Quels sont les paramètres et les types de retour de ces fonctions ? Ce type de générateur est une fonction déterministe qui utilise un germe. Si le germe n'est pas connu, la suite apparaît aléatoire. Pour initialiser ce germe, on utilise la fonction `srand` et ensuite à chaque fois qu'on veut une valeur aléatoire, on appelle la fonction `rand`. Comme `rand` renvoie des valeurs sur 32 bits, pour ne pas avoir des valeurs trop grandes, on évaluera `rand() % 100` pour tronquer le résultat sur 2 digits.

Enfin, si on ne veut pas que l'utilisateur rentre la valeur du germe, on peut utiliser diverses techniques. Une solution est de prendre comme germe la valeur du processus. Sous Unix, on utilise la fonction `getpid`. Quels sont les includes nécessaires et les paramètres de cette fonction ?

3.a] Faire de même avec un tableau initialisé avec des valeurs aléatoires.

Exercice 4 : Tri à bulle

Le principe de ce tri est d'effectuer plusieurs passes sur le tableau. À la i -ième passe, le tri met en place le i -ième plus grand élément en comparant les éléments en position $T[k]$ et $T[k+1]$ et les échange si besoin pour $k = 1$ à N . Cette borne N pourra être modifiée une fois qu'on aura bien compris le fonctionnement de l'algorithme. En effectuant N passes, on est certain qu'à la fin tous les éléments seront à leur place.

4.a] Programmer le tri à bulle.