
TD 10 : Standard Template Library

Compilation des exercices :

make ex1 ou *make ex2* ou ... ou *make ex11*.

make clean ARG = ex5 par exemple.

make all et *make cleanall*

Exercice 1 : Iterator

Ecrivez un programme qui crée un vecteur *v1* contenant les entiers 1, 2, 3, 4, 5. A l'aide d'un itérateur balayez le vecteur et affichez élément par élément. Faites de même en sens inverse en utilisant l'itérateur adéquate.

Nous allons maintenant coder l'algorithme sort pour rechercher la valeur *val* = 4 dans le vecteur *v1* en utilisant des itérateurs. Déterminez la position de *val* dans *v1*. N'oubliez pas la condition où *val* n'est pas trouvé. Essayer désormais avec *val* = 10. Que se passe t'il ?

Exercice 2 : vector

Écrivez un programme qui crée un vecteur *v1* de 10 float régulièrement répartis entre 0.0 (inclue) et 1.0 (non compris), complétez la fonction *showvec* pour affichez le vecteur.

Créez un nouveau vecteur *v2* égale à *v1* et effacez élément par élément la deuxième moitié de *v2*. Affichez *v2*.

Exercice 3 : List

Ecrivez un programme qui crée une liste *c1* contenant les 10 premières lettres de l'alphabet, complétez la fonction *showlist* pour afficher la liste.

Créez une nouvelle liste *c2* égale à *c1* et effacez d'un bloc la première moitié de *c2*. Affichez *c2*.

Exercice 4 : Deque

Ecrivez un programme qui insère à la fin du deque l'entier 10, en tête 20, à la fin 30 et en tête 15. Complétez la fonction *showdq* pour afficher le deque.

Renseignez la taille du deque, la valeur en 2ieme position, en tête du deque, à la fin du deque. Enfin supprimez le 1ier élément, affichez le deque, et supprimez la dernière valeur et affichez de nouveau le deque.

Exercice 5 : File ou Queue

1. Ajoutez les nombres 1, 4, 2 dans la file
2. Affichez la taille de la file
3. En faisant défiler les éléments de la file, affichez l'élément le plus ancien et supprimez le.
4. Affichez de nouveau la taille de la file

Exercice 6 : Pile ou Stack

1. Ajoutez les nombres 1, 4, 2 dans la pile
2. Affichez la taille de la pile
3. En faisant défiler les éléments de la pile, affichez l'élément en haut de la pile et supprimez le.
4. Affichez de nouveau la taille de la pile

Exercice 7 : Map

1. Remplissez la table associative (map) : mois et nombre de jours du mois
2. Affichez la taille de la map
3. Utilisez un itérateur pour parcourir la map et affichez les mois avec le nombre de jours correspondant
4. Affichez le nombre de jours du mois de Janvier
5. Affichez les jours du mois de Janvier

Exercice 8 : String

1. Initialisez s1 par la phrase : "coucou, dit le coucou au long cou."
2. Recherchez l'occurrence 'le' dans s1
3. Recherchez les occurrences 'cou' dans s1
4. Remplacez les occurrences 'cou' dans s1 par 'do' et affichez s1

Exercice 9 : Tri d'un tableau

1. Déclarer un vecteur d'entier initialisé à 1, 5, 2, 4, 3
2. Affichez les éléments du tableau
3. Triez dans l'ordre croissant le tableau à l'aide de la fonction sort
4. Triez dans l'ordre décroissant le tableau à l'aide de la fonction sort
5. Affichez de nouveau les éléments du tableau

Exercice 10 : Recherche d'un élément

1. Initialisez le tableau tab avec les entiers suivants 10, 20, 30, 40, 50 et recherchez l'élément 30 avec la fonction find
2. Initialisez le vector vec avec les valeurs suivantes : 10, 20, 30 et recherchez l'élément 30, déterminez sa position en utilisant l'itérateur itV
3. Initialisez la liste myList avec les valeurs suivantes : 5, 19, 34, 3, 33 et recherchez l'élément 34, déterminez sa position en utilisant la fonction distance et l'itérateur itL

Exercice 11 : Suppression d'éléments

Ecrivez un programme en initialisant le vecteur d'entier vec=0,1,2,3,4,5,6,7,8,9 dans lequel vous :

1. Supprimez le premier élément du vector, affichez la nouvelle liste d'éléments en utilisant la fonction "emplaté" showvec que vous complétez
2. Supprimez les éléments suivants : $[debutListe + 2; debutListe + 5]$, affichez le vector
3. Supprimez tous les éléments, affichez de nouveau le vector