

TP : à la découverte de la STL

Objectifs/Instructions

Découvrir certaines caractéristiques de la lib STL. Le TP est long, mais extrêmement dirigé au début ...

Avant de commencer, récupérez sur le compte polytech de Laure Gonnord le fichier `listes.cpp`. (`~lgonnord/ima3/PA/listes.cpp`).

1 Listes d'entiers, listes de chaînes de caractères

1. Ouvrir et observer le fichier `listes.cpp`, et ouvrir la documentation STL pour les listes dans un navigateur : <http://www.sgi.com/tech/stl/List.html>. Comment sont implémentées les listes dans la STL ?
2. Compiler en utilisant `g++` à la place de `gcc`. Exécuter.
3. Copier `listes.cpp` en `listes2.cpp` et modifier le fichier `listes2.cpp` pour gérer des listes de chaînes de caractères (`string`). Modifier aussi les tests. Compiler et exécuter.

► **Lorsque que le type support de la liste (ici : entier ou chaîne) possède une fonction de comparaison, les fonctions `sort` et `lower_bound` l'utilisent.**

Et si je veux trier par ordre décroissant ? La fonction `sort` peut utiliser une fonction de comparaison donnée par l'utilisateur.

1. Réouvrir `listes.cpp`
2. Regarder dans la doc `STL/Lists` (tout en bas) et écrire une fonction de comparaison sur les entiers compatible avec l'utilisation de `sort`.
3. Utiliser `sort` pour trier la liste d'entiers par ordre décroissant cette fois.
4. Vérifier en imprimant.

Quelques autres fonctions utiles Toujours sur le fichier `listes.cpp` (on pourrait faire de même pour `listes2.cpp` ...) :

1. Utiliser les fonctions de librairie pour concaténer deux listes d'entiers (listes non triées)
2. Utiliser un itérateur inverse pour imprimer une liste dans l'ordre inverse.

2 Le type Vector

Ce sont des tableaux proches des tableaux C, leur utilisation est facilitée par le fait qu'il y a réallocation automatique si il y a besoin. Nous allons tester dans un nouveau fichier `vector.cpp`.

1. Ne pas oublier les include :

```
#include <vector>
#include <iostream>
```

2. Dans le `main`, déclarer un vecteur d'entiers simple, et un vecteur d'entier de taille 5.
3. Écrire une fonction d'impression de vecteur d'entier qui utilise un itérateur (s'inspirer du modèle `listes.cpp`)
4. Écrire une fonction d'impression qui utilise les index du tableau (comme en C).
5. Insérer des éléments dans ces vecteurs, de deux façons (`push_back` et accès direct comme en C). Essayer d'insérer un élément à une case qui n'existe pas.
6. Tester et faire des conclusions.

3 Le type Map

Une map permet d'associer une clé (identifiant) à une donnée (table associative). Aux numéros 1, ..., 12 on associe les noms de mois "janvier" ... "décembre".

- Déclarer et remplir une telle map.
- Peut-on associer deux valeurs à la même clef ? tester !
- Parcourir et imprimer cette table.
- Créer la map inverse.
- Imprimer les mois par ordre alphabétique.