

# Exercices de C++ - jour 3

## La généricité :

### Exercice 0 :

Créez une classe générique appelée Liste qui peut stocker une liste de n'importe quel type de valeur. La classe doit avoir les fonctionnalités suivantes :

- Un constructeur
- Une méthode pour ajouter un élément à la liste.
- Une méthode pour supprimer un élément de la liste.
- Une méthode pour afficher tous les éléments de la liste.
- Une méthode pour vider tous les éléments de la liste.

### Exercice 1 :

Créez une classe générique appelée « Pair » qui peut stocker une paire de deux éléments de n'importe quel type. Ensuite, créez une classe dérivée « LabeledPair » qui hérite de la classe « Pair » et ajoute un label pour décrire la paire. Testez votre code.

### Exercice 2 :

Créez une classe générique appelée « HashTable » pour gérer une table de hashage. Cette classe doit contenir une collection de type `std::unordered_map` et doit posséder des méthodes permettant :

- d'ajouter un élément
- de rechercher un élément par clé
- de vérifier si la collection contient une clé particulière.

Créez ensuite une autre classe possédant une spécialisation partielle pour gérer spécifiquement des tables de hachage où les clés sont des chaînes de caractères et les valeurs des pointeurs. Testez ensuite vos classes.

### Exercice 3 :

Proposez une spécialisation totale de la table de hachage ci-dessus.

## **Introduction à la STL :**

### **Exercice 4 :**

Créez une classe Produit avec un id, un nom, un prix et une quantité. Créez une classe permettant de stocker les produits sous forme de liste chaînée. Cette classe disposera de méthodes permettant de :

- Ajouter des produits,
- Supprimer un produit selon son id,
- Afficher la liste des produits,
- Rechercher un produit par son nom,
- Afficher un produit spécifique en fonction de son id
- Mettre à jour la quantité d'un produit spécifique
- Trier la liste des produits par nom ou par prix.

### **Exercice 5 :**

Créez un vecteur de nombres entiers. Utilisez des algorithmes de la STL pour :

- Trier les nombres du vecteur par ordre croissant.
- Créer un nouveau vecteur qui contient le carré des nombres du vecteur original.
- Calculer la somme de tous les nombres du vecteur.
- Fusionner deux vecteurs de nombres aussi.
- Supprimer tous les nombres pairs du vecteur.
- Trouver un nombre dans un vecteur.
- Comptez tous les éléments d'un vecteur.
- Copier le contenu du vecteur dans un autre vecteur.

Enfin affichez les résultats obtenus.

### **Exercice 6 :**

Reprenez l'exercice 4 et créez une classe manipulant des fichiers permettant de stocker un produit sous forme de fichier et de stocker la liste des produits dans un fichier.