

1. Quel est l'avantage d'utiliser une linked-list au lieu d'un vecteur si on veut supprimer ou rajouter un élément ? Quelle structure de données est plus avantageuse si on veut accéder à des éléments à des positions aléatoires ?

Une liste liée permet d'ajouter/supprimer des éléments au début ou au milieu sans avoir à décaler toutes les valeurs comme il faudrait le faire avec un vecteur. L'opération se résume à 2 affectations de pointeurs plutôt, ce qui est plus rapide. Cependant, pour trouver l'endroit d'insertion, il faut parcourir la liste à partir du début. C'est pourquoi les vecteurs sont plus avantageux pour accéder à des éléments à des positions aléatoires, cela se fait grâce à l'opérateur [].

2. Pourquoi est-ce que l'implémentation des classes génériques est dans .h et non pas séparée en .h et .cpp comme les classes normales ?

Le compilateur doit avoir accès à la définition et à l'implémentation pour produire une version spécifique de la classe. Comme par exemple, la classe GestionnaireCoupons est définie dans gestionnaireGenerique.h et implémentée dans gestionnaireCoupons.cpp.