

Question

Meteo

On a une map de string pour des paires de double (température et vent). Une `unordered_map` serait mieux pour l'insertion et le retrait ($O(1)$ amortie).

Frigo

Dans le frigo, on peut imaginer avoir plusieurs fois le même aliment. On a souvent de l'ajout/-retrait, en $O(1)$ avec une liste, donc le conteneur choisi est approprié.

Routeur

On a une map d'id pour des pointeurs d'objets connectés. Une `unordered_map` serait mieux pour l'insertion et le retrait ($O(1)$ amortie).

Personne

On a une liste d'allergie (string). Ce n'est clairement pas optimal, un set serait bien plus approprié. De cette façon, on est sûr de ne pas avoir de doublon et la recherche d'allergie serait en $O(\log(n))$. L'ajout d'allergie serait plus couteux en $O(\log(n))$, mais cela en vaudrait la peine vu que c'est une opération qui sera « rare » après la construction d'une personne (on ne devient pas allergique à quelque chose tous les jours...). En plus, pas d'opérateurs à redéfinir (opérateur `string <` par défaut).

Maison

On a une liste de personnes (pointeurs sur personnes). C'est pas mal, mais là encore, un set serait bien plus approprié : les doublons n'ont pas de sens. De plus, les opérations courantes sur les sets nous permettraient de faire tout plein de choses folkloriques (union, intersection, différence...), qui pourraient s'avérer très utile (déménagement, nouveaux locataires...). En plus, pas d'opérateurs à redéfinir (type pointeur).