

TP2 - Définitions complètes de classes

Pour chaque sous-programme ou une méthode, définir son jeu de tests et préparer les tests automatiques (pour les fonctions/méthodes ne faisant pas d'entrée/sortie) dans le fichier `main.cpp`.

Introduction

Une agence immobilière désire modéliser le portefeuille des appartements dont elle assure la location. Un `Appartement` sera simplement composé d'un ensemble de `Pièces`.

Partie 1

1. Déclarez une classe `Piece` : une pièce est définie par ses deux dimensions (sa largeur et sa longueur, en mètres) et son nom (en minuscules, par exemple `"sejour"`).
2. Écrivez un constructeur prenant 3 paramètres pour initialiser les attributs, et un constructeur vide initialisant une pièce de 3m x 4m nommée `"piecevide"`. Ne pas affecter de valeurs par défaut aux paramètres.
3. Ajoutez une méthode `afficher` qui permet d'afficher les caractéristiques de la pièce.
4. Modifiez un des deux constructeurs (réfléchissez à celui qui doit être modifié) de manière à avoir une pièce bien structurée dès la création de l'objet :
 - la longueur doit être supérieure ou égale à la largeur (permuter si ce n'est pas le cas) ;
 - les dimensions doivent être strictement positives (déclencher une exception si ça n'est pas le cas) ;
 - le nom de la pièce doit être en minuscule (faire les modifications nécessaires).
5. Ajoutez une méthode `saisir` qui permet de demander à l'utilisateur les caractéristiques de la pièce. Penser aux vérifications.
6. Ajoutez une méthode `surface` qui renvoie la superficie d'une pièce.
7. Ajoutez une méthode `plusGrande` qui, recevant une pièce en paramètre, renvoie `true` si la superficie de la cible est plus grande que celle de la pièce passée en paramètre.
8. (optionnelle) Définir une autre fonction `surface` externe à la classe `Piece` (elle peut porter le même nom).

Partie 2

1. Déclarez une classe `Appartement` : un appartement est caractérisé par son adresse (une chaîne de caractères) et l'ensemble des pièces qui le composent (un vecteur de `Pièces`).
2. Ajoutez un constructeur vide à cette classe, construisant un appartement vide. Il aura une adresse par défaut, par exemple `"1 rue Joliot-Curie 91400 Orsay"`.
3. Ajoutez un second constructeur qui construit un appartement avec une seule pièce passée en paramètre, et une adresse passée en paramètre.
4. Ajoutez une méthode `afficher` qui affiche les différentes pièces d'un appartement.
5. Ajoutez une méthode `ajoutPiece` qui ajoute une pièce passée en paramètre.
6. (optionnelle) Ajoutez une méthode `retraitPiece` qui, étant donné un indice passé en paramètre, supprime la pièce enregistrée à cet indice et renvoie la pièce ainsi supprimée. La pièce supprimée est remplacée par la dernière pièce.
7. Ajoutez une méthode `nbPieces` qui renvoie le nombre de pièces d'un appartement.
8. Ajoutez une méthode `surface` qui renvoie la surface totale d'un appartement.
9. Ajoutez une méthode `compare` qui, recevant un appartement en paramètre, renvoie `-1`, `0` ou `1` selon que l'appartement cible a une surface plus petite, égale ou plus grande que l'appartement passé en paramètre.
10. (optionnelle) Ajoutez une méthode `plusGrandePiece` qui renvoie l'indice de la pièce de plus grande surface (`-1` si aucune `Piece` dans l'appartement).

Partie 3

La valeur d'un appartement est estimée en prenant en compte le prix au m² de la zone concernée. Il y a cinq zones (numérotées de 0 à 4) dont la valeur du m² est la suivante :

Zone	0	1	2	3	4
Valeur (€)	2500	3100	3500	4150	4500

?

1. Modifiez la classe `Appartement` en ajoutant un attribut `_zone`, qui contiendra le numéro de la zone dans laquelle se situe l'appartement. Adaptez en conséquence les constructeurs de cette classe et toutes les méthodes concernées. Pensez à utiliser des constantes (déclarées et définies dans le fichier `.h`).
2. Écrivez une méthode `valeur` renvoyant la valeur estimée de l'appartement.
3. Ajoutez une méthode `comparePrix` qui, recevant un appartement, renvoie `-1`, `0` ou `1` selon que l'appartement cible a une valeur plus petite, égale ou plus grande que l'appartement passé en paramètre.

Modifié le: mardi 28 novembre 2023, 15:41