

Programmation orientée objet (C++) – ING2-GSI

TP8 : patron observateur

Exercice 8.1 : Point observable

Nous utilisons le patron de conception « observateur » pour résoudre le problème de dépendance entre les segments et les points. Un objet point gère une liste d'observateurs qu'il avertit (en appelant l'opération maj, mettre à jour) lorsqu'il est traduit.

Un segment dont la longueur est stockée peut alors s'inscrire auprès de ses points extrémités pour que sa longueur soit recalculée chaque fois que l'un de ses points extrémités est traduit. La figure 1 contient le diagramme de classe correspondant à une solution possible. L'attribut observateurs de la classe GroupeObservateurs sera défini comme une collection. C'est la manière choisie pour représenter la cardinalité * de GroupeObservateurs avec l'interface Observateur. On utilisera la classe GroupeObservateurs pour réaliser la relation entre Point et Observateur. La classe Point aura donc un attribut observateurs de type GroupeObservateurs.

En respectant le diagramme de classe, proposer une implémentation pour ce problème.

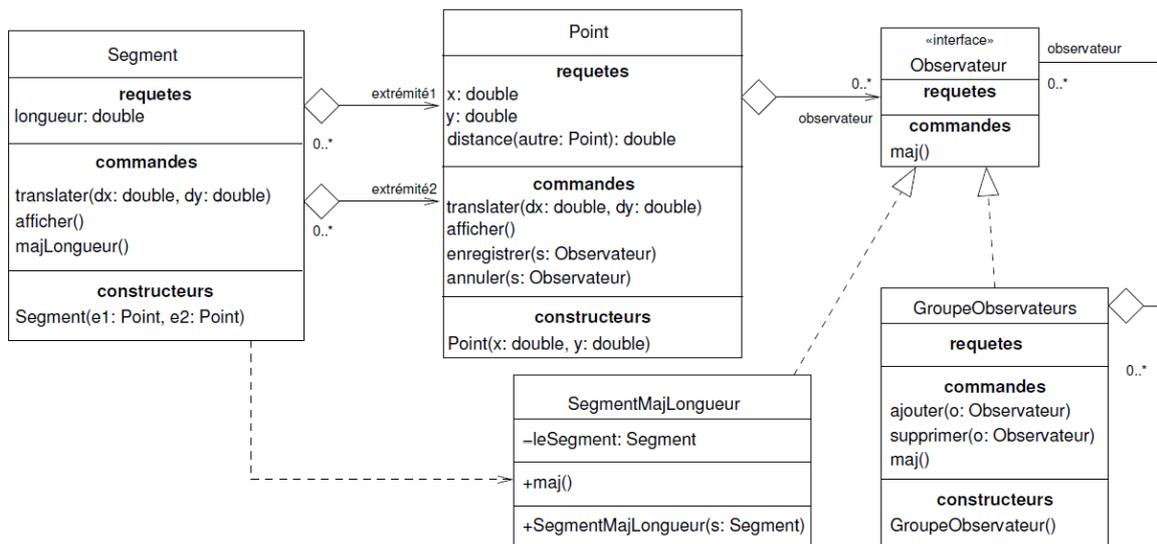


FIGURE 1 – Diagramme de classes des points observables.

Exercice 8.2 : Cercle et Point

Considérons maintenant un Cercle défini par son centre et un point de sa circonférence. On considère que le rayon du cercle est une information stockée et qu'il y a une relation d'agrégation entre le cercle et ses deux points. La translation de l'un de ces deux points a donc un impact sur le cercle :

- une translation du centre du cercle provoque une translation du point de la circonférence ;
- une translation du point de la circonférence change la taille du cercle (le centre reste inchangé) et nécessite donc de mettre à jour le rayon

Adapter les classes pour prendre en compte les cercles.