

L'équipe pédagogique	Ref : POO Java
A l'intention des étudiants d'INGMAN2	26 juillet 2017

Jeu d'échec

On s'intéresse ici au déplacement de quelques-unes des pièces d'un jeu d'échecs. L'idée principale est de représenter le jeu comme une collection de pièces (en général), puis de spécifier chaque type de pièce. Chaque pièce est caractérisée par une position et une couleur :

- La couleur peut être « blanc » ou « noir ».
- On gère la position comme une classe. Elle est définie par un couple de coordonnées (L, C) comprises chacune entre 1 et 8.
En plus des méthodes usuelles (constructeur(s), modificateurs, affichage, ...), à définir selon les besoins, une pièce aura une méthode *deplace* permettant de la déplacer vers une position donnée en argument. Cette méthode retournera un booléen « vrai » si le déplacement est valide (et s'est effectué) et « faux » sinon.

On souhaite ici implémenter 2 pièces d'échecs, le cheval et la tour :

- Le cheval se déplace d'une case dans une direction et de deux cases dans l'autre (en avant ou en arrière). Par exemple si le cheval est en (4,5), il peut aller en (5,7), (5,3), (3,7), (3,3), (6,6), (2,6), (6,4) ou (2,4).
- La tour peut se déplacer d'autant de cases qu'elle veut, mais toujours dans la même ligne ou dans la même colonne. Par exemple si une tour est en (4,5), elle peut aller en (4,1), (4,2), ..., (4,8), (1,5), (2,5), ..., ou (8,5) (mais ne peut pas « se déplacer » en (4,5)).

Pour terminer la modélisation, on représentera le jeu comme une classe avec simplement une collection de pièces et une méthode d'affichage donnant la liste des pièces qu'il contient (on ne s'intéresse pas ici à l'affichage complet du jeu sous forme de grille).

Questions :

1. Après les avoir définies, expliquer la différence entre une agrégation et une composition.
2. Quel est le lien entre la classe Pièce et les autres pièces d'échecs (cheval, tour, ...) ?
3. Construire le diagramme de classe UML de ce jeu.
4. Écrire en Java les classes Position, Piece, Tour et Jeu comme décrites ci-dessus.
5. Écrire un programme de test permettant de réaliser les opérations suivantes :
 - ✚ Construire dans le main un jeu contenant : une tour noire en (7,6) et une tour blanche en (3,2), puis afficher le jeu.
 - ✚ Déplacer la tour blanche en (7,1), la tour noire en (7,1) et afficher le jeu.

Bonus : implémenter la classe Tour.

NOTE : On ne se préoccupe pas dans ce programme de savoir si la case d'arrivée est occupée ou non, ni si la trajectoire est « encombrée ». On ne s'intéresse ici qu'aux déplacements de base.