

## Exercices de Patterns (7)

### Question examen 2014

Réalisation de patterns Vous allez réaliser une partie d'un moteur de combat entre des robots. Chaque robot est caractérisé par :

- Des points de vie. Quand il n'y en a plus, le robot explose et le combat se termine. Par défaut, les robots ont 100 points de vie.
- La puissance d'un canon. Chaque fois que le robot tire, c'est la puissance de ce canon moins les défenses de l'adversaire qui est supprimé des points de vie du robot adverse. Par défaut le canon a une puissance de 1.
- La puissance d'un bouclier. Ce dernier enlève une valeur figée des dégâts infligés. Par défaut le bouclier a une puissance de 1.
- Une fréquence de tir. Plus la valeur est élevée, plus le robot prendra du temps pour tirer. Par exemple une fréquence deux fois plus petite permet à un robot de tirer deux fois plus souvent. Par défaut, la fréquence de tir est de 100.

Deux fichiers vous sont fournis :

- PatternRobots.java : vous devez uniquement modifier le main dans ce fichier, pas les autres méthodes.
- Robot.java : vous ne devez rien changer à ce fichier. Lisez bien par contre la documentation des méthodes.

Il faut un mécanisme permettant de configurer les valeurs par défaut uniquement à la construction du robot. Une fois construit, ces valeurs sont immuables. De plus il faut offrir la possibilité de ne redéfinir que les valeurs que l'on souhaite, en utilisant les valeurs par défaut au sinon. Quel pattern allez-vous utiliser ?

*Participants théoriques*

*Noms utilisés dans le code*

*Nom théorique des méthodes*

*Nom utilisé dans le code*

## Exercices de Patterns (7)

De plus le jeu permettra d'obtenir des améliorations permettant de changer le comportement du robot. Une de ces améliorations permettra de multiplier par deux les dégâts du canon, une autre de multiplier par deux la puissance de bouclier, une autre encore de diminuer par deux la valeur des points de vie perdus, etc... Vous devez donc trouver un mécanisme permettant de reconfigurer à volonté le comportement des robots tout en gardant l'immuabilité des attributs. Quel pattern allez-vous utiliser ?

*Participants théoriques*

*Noms utilisés dans le code*

*Nom théorique des méthodes*

*Nom utilisé dans le code*

Finalement, il serait pratique d'avoir un mécanisme permettant d'obtenir des robots préconfigurés. Par exemple, obtenir un robot « pic-vert » qui a peu de puissance d'attaque et de défense, mais tire très fréquemment. Un autre, la « grosse berta » qui par contre est très lent, mais fait beaucoup de dégâts à chaque tir, ou encore un autre, le « tank » qui est lent, fait peu de dégâts, mais a beaucoup de points de vie et reçoit très peu de dégâts. Créez un tel mécanisme en le remplissant des 3 exemples indiqués ici. Quel(s) pattern(s) allez-vous utiliser ?

Pattern 1 :

*Participants théoriques*

*Noms utilisés dans le code*

*Nom théorique des méthodes*

*Nom utilisé dans le code*

Pattern 2 :

*Participants théoriques*

*Noms utilisés dans le code*

*Nom théorique des méthodes*

*Nom utilisé dans le code*