

## Exercice 06

---

### Jeu d'Othello

Programmation du jeu d'Othello

1. ENONCE 2
2. CORRECTION 2

## 1. Enoncé

On se propose de réaliser le jeu d'othello.

Les règles sont expliquées : <http://www.ffothello.org/jeu/regles.php>

Chaque joueur a son propre plateau de jeu.

On affiche deux ihms , une pour chaque joueur.

Les deux ihm peuvent s'afficher sur le même écran, localement, dans un même programme, pour faciliter les tests. Mais, bien sûr, il est possible d'afficher les deux ihms sur 2 machines différentes. Les 2 programmes communiquent entre eux à travers le réseau.

Ce programme doit se faire suivant le modèle MVC.

Dans le cas où les 2 joueurs jouent en réseau, le programme d'un joueur est le maitre et l'autre l'esclave dans lequel la Vue est un client du MVC du programme maitre.

Les Designs Patterns que vous allez utiliser doivent permettre facilement le passage entre un fonctionnement local ou distant.

Il doit être également possible de remplacer, facilement, un des utilisateurs humains (ou les deux) par un robot qui joue automatiquement au jeu d'othello.

Faire les diagrammes de classes de conception d'un tel programme.

## 2. Correction

En orange le diagramme de classe correspondant au programme dans lequel les vues sont toutes deux locales et appartiennent à la même JVM (modèle MVC classique).

En vert les classes ajoutées pour que les 2 joueurs puissent jouer en réseau.

Les classes dont les noms sont en rouge correspondent aux classes utilisées par le joueur 2, programme esclave.

En blanc les classes permettant de créer un robot qui joue automatiquement au jeu d'Othello.

Le programme esclave utilise un contrôleur distant **ControleurOthelloDistant** qui sert de proxy de communication avec le contrôleur OD **ControlleurOthelloODImpl** qui sert d'adaptateur au contrôleur.

L'esclave ne contient pas le modèle et l'observable.

La classe **ProxyObserverAsynchrone** est utilisé pour que la notification à chaque observer soit fait à travers un thread. Cette classe est indispensable pour l'utilisation des robots qui jouent à la place d'un ou des deux joueurs.

Le robot **RobotOthello** est un observer qui joue sur notification d'un nouvel état du modèle et si c'est à lui de jouer.

Le robot joue avec un algorithme heuristique de type Alpha-Beta dont la fonction d'évaluation consiste à rendre prioritaire un coup dans chacun des 4 coins, à donner une

