Labo 08– Echecs

Auteurs : P.Gillet & L.Haye

Groupe n° L08GrC

Table des matières

[1. UML 1](#_Toc155688308)

[2. Choix de conception 1](#_Toc155688309)

[3. Tests 1](#_Toc155688310)

# UML

# Choix de conception

* Chaque pièce du jeu découle de la classe abstraite Piece.
* Afin de factoriser un peu notre code, nous avons représenter les coordonnées x, y d’une case avec la classe Point, disponible dans java.awt.
* Avant d’effectuer un mouvement, une pièce doit vérifier si ce déplacement est autorisé. C’est ensuite que le mouvement est effectué sur le plateau de jeu.
* Afin de factoriser notre code, nous avons fait en sorte que les pièces ayant le même type de mouvement (linéaire, diagonale et/ou restreint par une distance) utilise la même interface.
* Pour vérifier que le mouvement d’un joueur ne le mette pas en échec lui-même, nous effectuons ce mouvement avec une pièce factice puis vérifions si le roi est sain et sauf. En cas d’échec et mat, il ne sera donc plus possible d’effectuer de mouvement.

# Tests

Concernant les tests, nous testons ces cas :

+ Pour le plateau de jeu :

* Sa création
* Sa réinitialisation
* Le changement de couleur après chaque tour
* Le getter pour des cases
* Le placement d’une pièce
* Le déplacement d’une pièce
* Le déplacement d’une pièce en cas d’échec
* La suppression d’une pièce
* Les mouvements spéciaux (roque, prise en passant, promotion)
* La dernière pièce jouée
* La définition du nombre de tour
* La position de départ de chaque pièce

+ Pour chaque type de pièce :

* Les déplacements propres à la pièce et autres cas spéciaux
* Le getter pour son type
* Le getter pour son nom
* Sa méthode toString.