Project 5 Testing Document

Nobel Zhou (nxz157)

**6 December 2020**

Introduction

This the Project 5 testing document for all methods and classes. This chess program represents a playable game of chess, including promotion, check, checkmate, stalemate, draw by threefold repetition, and draw by 50 move rule. All moves are checked to make sure they don’t result in check. In addition, the chess board has been updated to add piece icons instead of just the symbol and the available moves are also highlighted when you click on the piece. There are a lot of methods that are unnecessary in the light of grading for this program but deemed necessary for the functionality and accuracy of creating a chess program. In addition, I will **not** be testing any methods that were pre-coded. I will be testing interface and abstract class methods through a concrete class because they are not available to be tested through their instances because they don’t have one. This testing document will be broken down and tackled within each class file. In addition, I have attached images containing the chess positions and the expected results of the resulting position if Junit is not feasible for testing those types of interactions. All chess board visualizations were taken using the analysis board at pychess.com/editor/xiangqi. Also, the testing uses the TestChessBoard class to test all methods related to the pieces and the gamerules so the GUI doesn’t get in the way of things. I will also **not** be testing any methods that relate to Indo-European chess. I will **not** be Junit testing any GUI interface as well, but instead describe the visual tests and interaction pane code that I did to achieve the desired result.

XiangqiKingPiece, CanPalaceMove, CanFaceKingMove, and CanSingleStraightMove

## Introduction

This will cover all methods in the XiangqiKingPiece class as well as its implemented interfaces. I will first check instantiation of a XiangqiKingPiece, and then check its move methods.

XiangqiKingPiece::XiangqiKingPiece()

For the first test, I will check the instantiation of a XiangqiKingPiece object by calling its constructor and then assigning it to a variable. Then I will call the getLabel method to make sure that the correct piece was created.

The test is grouped under testXiangqiKingPiece() in XiangqiChessTester.java::000-000. The test passed successfully.

XiangqiKingPiece, CanPalaceMove, CanFaceKingMove, and CanSingleStraightMove

## Introduction

This will cover all methods in the XiangqiKingPiece class as well

StraightMove::checkEmptyStraightMove()

For the first test, I

The test is grouped under testMethod() in XiangqiChessTester.java::000-000. The test passed successfully.