Milestone 1

CPSC 5210-01
Group 6: Khanh Phan, Shaista Usman, Yi Niu

I. Project Description:

The source project chosen by our group is a classic game called Tetris written in Java.
In this game, a random block of different shapes will appear from time to time, allowing the user to rotate the blocks, move left, move right or move down. The goal is to fill as many lines of the grid as possible to earn points. Everytime the user makes a move, the program will actively check to make sure it's not blocked by the current state of the grid before executing the user's action.
We use Eclipse and IntelliJ IDEs as well as J-Unit framework for our test suite.

Github URL: https://github.com/Glank/Java-Games/tree/master/Tetris

II. Test Instructions:

Requirements:

- Eclipse IDE for Java
- JUnit 4 (built in Eclipse IDE)
- JDK

Project download and import:

- Git clone the repository to local disk: https://github.com/IrenaNiu/Java-TetrisTest.git
- Open IDE of Eclipse and import the project into the IDE. File --> Open Projects from File System... --> Select directory --> Click OK
- The JUnit 4 library could be set up by BuildPath to make sure the JUnit tests are working.

Instructions:

Run the game application in Eclipse:

- Open the package of "tetris" and right click on "PlayTetris.java"
- Run as "Java Application"
- You will see the Tetris game loading with a window to play. Left and right to move. Up to rotate.

Run the unit tests Eclipse:

- Open the package of "unitTests" and right click on the any of the three test java files
- Run as "JUnit Test"
- You will see the tests successfully pass within a second.

The units we are testing are “TetrisGrid,java”, “TetrisBlock.java” and “BlockGrid,java”. They have move functions for playing the blocks, getter and setters for block and grid attributes, and validation functions for return booleans.

Github URL: https://github.com/IrenaNiu/Java-TetrisTest

III. **Test Results:**

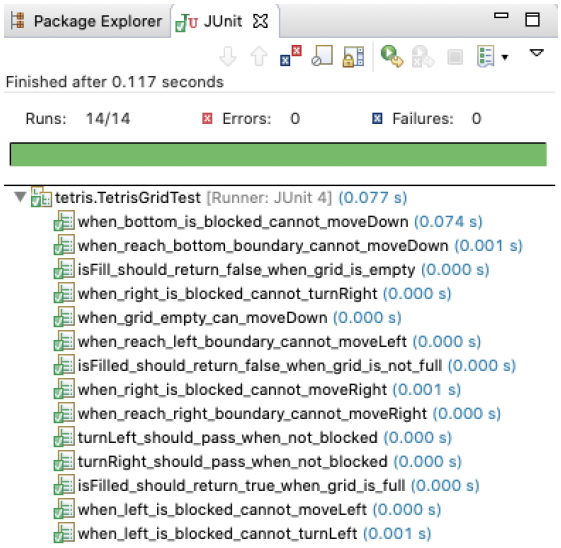

Figure 1. JUnit Test Run Results for TetrisGrid

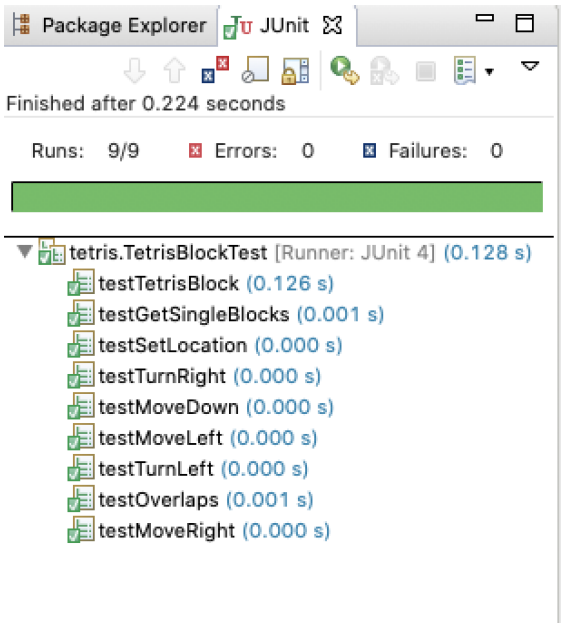

Figure 2. JUnit Test Run Results for TetrisBlock

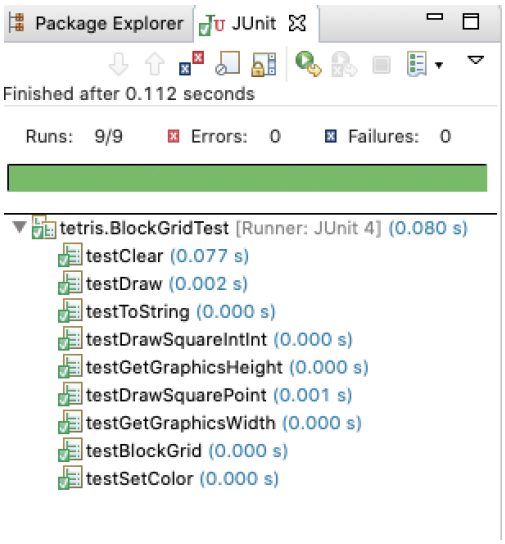

Figure 3. JUnit Test Run Results for BlockGrid

IV. Experience:

We decided to work on java projects as most of us were comfortable with Java as a programming language and found JUnit to be fairly easy to understand and work on. We had picked out three open source projects as potential units for this first milestone. One of the choices “Cabin Town”, a text game, was testable, however had very few classes and was too simple to be considered for this milestone. Both of our other two choices, “Chess” and “Tetris” (Java games) had dependencies in the form of GUI components, however on close examination we found that these had classes and methods that did not have a lot of

dependencies, where the codes were properly segmented and displayed repeatable events. We chose to go ahead with Tetris as Chess had seemed to be too big of an undertaking in this context.

We were able to import and build Tetris without any difficulty. We were already aware of the basic premise of the game and with some effort and teamwork, we were able to further understand the underlying code of this implementation of the game. Since Tetris has a few GUI dependencies, we were unable to test such components. However, we were able to come up with unit tests for most of the functionality. Additionally, we had to include a few getter and setter methods for private class members in the original code, to make testing feasible for functions accessing and utilizing these private members. In conclusion, we were able to apply testing and debugging concepts learned from the class and also gain hands-on experience of writing unit tests for a small nonetheless functioning project.