README: Classic Platformer - Version 0.0.3

GITHUB LINK: https://github.com/johnhutchinson99/classicplatformer

This is a classic platformer game that is being created for a computer science course project.

In this game, your character is placed in a world map and must avoid enemies/obstacles in order to reach a final destination.

In the current iteration of this project, the game is in its second GUI demo stage.

Getting Started: Accessing the Demos

These are the instructions for accessing Demos 1, 2, 3 of our project, the text-based and GUI-based version of our game respectively.

Pre-requisites

To compile and run this project, you must haveJava SE Development Kit 8 installed.

Accessing Demo 1 - Compiling and Running: Using Eclipse

1. Download the repository (Version 0.0.1) as a ZIP file (on Github.com, choose "Clone or download" >> "Download as ZIP") and unzip on your computer.
2. Open an Eclipse workspace.
3. Within Eclipse, go to "File" >> "Open Projects from File System".
4. In resulting the pop-up, click "Directory" to choose the "Import Source" as the project folder you unzipped in Step 1. The name of the folder should be "classicplatformer"
5. Now that the project is imported into Eclipse, run the project by right-clicking "Gameplay.java" in the Eclipse Package Explorer and choosing "Run As" >> "Java Application".

Accessing Demo 1 - Compiling and Running: Using Command Line

1. Download the repository (Version 0.0.1) as a ZIP file (on Github.com, choose "Clone or download" >> "Download as ZIP") and unzip on your computer.
2. Open command line and set your present working directory to the src directory, which is found within the folder you unzipped in Step 1.
3. Run "javac *.java"
4. Run "java Gameplay"

Accessing Demo 2 - Compiling and Running: Using Eclipse

1. Follow Steps 1-4 of from "Accessing Demo 1 - Compiling and Running: Using Eclipse", except download Version 0.0.2 in Step 1.
2. Now that the project is imported into Eclipse, run the project by right-clicking "GUI.java" in the Eclipse Package Explorer and choosing "Run As" >> "Java Application".

Accessing Demo 2 - Compiling and Running: Using Command Line

1. Follow Steps 1-3 of from "Accessing Demo 1 - Compiling and Running: Using Command Line", except download Version 0.0.2 in Step 1.

2. Run "java GameGUI"

Accessing Demo 3 - Compiling and Running: Using Eclipse

3. Follow Steps 1-4 of from "Accessing Demo 1 - Compiling and Running: Using Eclipse", except download Version 0.0.3 in Step 1.
4. Now that the project is imported into Eclipse, run the project by right-clicking "GameGUI.java" in the Eclipse Package Explorer and choosing "Run As" >> "Java Application".

Accessing Demo 3 - Compiling and Running: Using Command Line

1. Follow Steps 1-3 of from "Accessing Demo 1 - Compiling and Running: Using Command Line", except download Version 0.0.3 in Step 1.
2. Run "java GUI"

Running Tests

Running JUnit Test in Eclipse

1. Follow Step 1 of "Accessing Demo 2 - Compiling and Running: Using Eclipse"
2. Ensure that the JUnit 4 library is included in the Java Build Path of the project.
3. In the Eclipse Package Explorer, right-click "PhysicsTest.java", hover over "Run As", then select "JUnit Test".

Running JUnit Test in Command Line

1. Follow Step 1 of "Accessing Demo 2 - Compiling and Running: Using Eclipse"
2. Download junit-4.12.jar and hamcrest-core-1.3.jar and place them in the src folder.
3. In the command line, change the working directory to the src folder and compile using "javac -cp .:junit-4.12.jar:hamcrest-core-1.3.jar *.java"
4. Run "java -cp .:junit-4.12.jar:hamcrest-core-1.3.jar org.junit.runner.JUnitCore PhysicsTest"