CSC207 Lab 7: Structural + Creational Patterns

Due: November 8, 8 p.m.

1 Introduction

This week we'll be working with the Factory Pattern (a Creational Pattern) and the Adapter pattern (a Structural Pattern). The Adapter pattern facilitates communications between objects that have differing, but relatable, interfaces. The Factory pattern will be used to simplify creation of a variety of objects that support the same interface. This week's lab is also special in that you may work on it with a partner, if you want! This will let you experiment with gitflow, which you will need to use for your group project, too.

To begin, copy the files from the course Quercus (in the 'lab07.zip' archive) into your local copy of your GitLab repository. After you have done this, you should have one folder named 'lab07' in your repository. You should find a folder called week7 in the lab07 folder and several java files therein, including: *FirstGenXbox.java, SecondGenXbox.java, XboxController.java, XboxControllerFactory.java, PSController.java, XBoxPSAdapter.java, and GTAVBackdoor.java.* You will also find a file containing tests.

2 Programming Task

You have two tasks this week:

1. Implement Adapter classes to facilitate adaptation of PSControllers cars to the the game of Grand Theft Auto (GTA). An adapted controller should be able to execute all the cheat (or back door) codes for the game! More specifically, you will need to write the adapter (*XBoxPSAdapter*) so that back door codes can be issued with a PSController. *XBoxPSAdapter* will therefore:

    - wrap a *PSController*
    - map any *XBoxController* key press onto a corresponding *PSController* key press. You can make an XBox 'A' key correspond to a PS 'X' key, an XBox 'B' key correspond to a PS controller's 'Circle' key, an XBox 'X' key map onto the PS controller's 'Square' key, and an XBox 'Y' key map onto the PS controller's 'Triangle' key. Also make an *XBoxController* reset equivalent to clearing a *PSController*. Note that there is no adaptation for the 'L' and 'R' buttons on an XBoxSecondGen controller, however.
    - return a state that is equivalent to an *XBoxController* state if and when any *GTAVBackdoor* cheat code is entered into a *PSController*.

In addition we ask that you implement an XBoxController Factory. This factory should:

- manufacture a First Generation XBoxController if and when the method $getController(”FirstGen”)$ is called.
- manufacture a Second Generation XBoxController if and when the method $getController(”SecondGen”)$ is called.

That’s it! We ask that you and your partner submit the same code to each individual repository. Remember to respect the organization of the starter code in your submissions!

3 What to Submit

1. XBoxControllerFactory.java
2. XBoxPSAdapter.java

HAVE FUN AND GOOD LUCK!