CS335 Program 0
"Memory"
Due Wednesday 9/20

1. Introduction

The goal of this assignment is to build a stand-alone Java program that will allow a user to play the *Memory* game. The graphical user interface should support the basic elements of the game: start, restart, play of the game itself, recording the number of guesses so far, and detecting the termination state. Note that your solution should be a stand-alone Java program, not an Applet.

2. The Game

Memory is played on a 4x6 board (4 rows and 6 columns). Each element of the grid represents a card. The game starts with all positions on the board blank, the "guesses made" counter at 0 and the "matches made" counter at 0. Play is as follows:

- The player makes a guess by clicking with the mouse on one grid element (card). The game should *toggle* that grid to reveal the pattern it hides (a unique image).
- The player makes a second guess by clicking on another square with the mouse. If the second square reveals a "match" with the first square, the squares are permanently displayed (remain unhidden and are no longer responsive to clicks) and the "matches made" counter is incremented
- If the two selected squares do not match, both are toggle back to the hidden state
- The "number of guesses" counter is incremented by 1
- The game ends when all grid elements are revealed.

Note that your game counter should count "guesses made" (it will be ½ the value of "clicks").

3. Implementation

Use the JButton element with the ability to support icons. You are responsible for choosing the icons for your game. Note that you will need 12 distinct icons, and each icon will be placed under two different places on the 4x6 board. The "start/restart" option, which initiates game play, should randomly associate the icons with buttons on the grid, so that the icons do not fall in the same location in every game.

4. What to Turn In

Turn in your IntelliJ Java project and all icons/files necessary to compile and run your code. Do this by zipping or using tar/gz to create a single archive from your IntelliJ project, and upload that archive to Canvas for this assignment.