Project 2 deliverables are worth 8% of your overall course mark.

Objectives
The purpose of this assignment is to:
1. Practice using the Model-View-Controller design pattern to separate your graphical user interface from the game model. This will also be useful in CS349, which relies heavily upon MVC.
2. Practice using the Facade design pattern to create a simplified interface and/or set of operations for your game logic.
3. Experiment with using an open-source C++ graphics library, gtkmm, which bears enough similarities to Java's graphical API that you'll be able to more easily pick up the material in CS349.

Due dates
The questions are due on Friday, July 14, 2017 at 11:59pm. They are to be submitted electronically through Marmoset.

Provided files
Some starter code and gtkmm examples have been provided in the ZIP archive, providedFiles.zip. Also provided is a sample executable, sampleGUIStraights. Yours may be quite different in appearance, but the basic functionality should be similar.

Marking rubric
We will grade your program based on its design and programming style as well as its correctness. See the rubric for details on how your program's design and programming style will be marked.

This deliverable will NOT be marked automatically, so the output specifications are less strict. Instead, you will demo your submitted solution for a TA in one of the CSCF labs. Your solution and demo MUST run on one of the linux.student.cs.uwaterloo.ca machines, so be sure that your solution doesn't use libraries that are not installed on these machines. Use UNIX command `locate gtk+` to check that the machine(s) you are using has GTK+ installed.

Project 2
You are to replace the text user interface from project 1 with a graphical user interface. If you have designed your project to follow the MVC design pattern, you should only need to change the interface i.e. no other parts of your project should be affected.

Submission
Submit your answer to Marmoset in a ZIP archive named p2.zip. It must be an archive of the directory, not an archive of the files as in previous assignments. The archive must include the files:

- [Mm]akefile creates an executable named straights when the command make is given,
- all .h files,
- all {.cc,C,cpp} files,
- all images, and
- the bonus features text file, bonus.txt.

Resources
There are a number of links on the Resources section of the course web page, https://www.student.cs.uwaterloo.ca/~cs247/current/resource.shtml. Included is a link to the official and unofficial gtkmm 3 tutorials and some notes about using it in the student environment. You will also find the provided examples helpful.