CMSC 202 — Fall 2013
Project 1 — Connect Four

Assigned	Monday, September 16thth
Program Due	Monday, September 30thrd by 8:00am
Weight	6%
Updates	All your code should be in a package called proj1

Connect Four

We will be programming an interface to simulate a connect four game. Those unfamiliar with connect four should look here for more details. The entire game will be texted based, with players taking turns entering moves.

Your connect four program should first prompt the user for a number of rows, then a number of columns. If either of these values is less than five, the program should continue to ask the user for a row or column value until the user inputs a number greater than four.

Once a number of rows and number of columns has been succesfully been entered, you should continue to prompt each player for a move in turn until a player has won or submits 'q' as a move. Once either player has won, you should present an option to play again by typing y, or to quit by typing n.

For example:

```
Hi, please enter a number of rows:  5
Please enter a number of columns:  5
_____
_____
_____
_____
_____
Player one, please enter a move: 1
_____
_____
_____
```

```
_____
x____
Player two, please enter a move: 1

_____
_____
_____
o____
x____
```

Please note that your program must follow this format exactly to get any credit. Programs that do not use these symbols and spacing will receive a grade of zero.

How to Start

We will be creating two files, Project1.java and ConnectFour.java. Project1.java will contain all the input and output code required to play the game; ConnectFour.java will contain the inner workings of the game. At no time should ConnectFour.java ever communicate with the user.

Project1.java

Project1.java will do all the input and output. It will need an instance of the ConnectFour object to communicate player's moves to. Remember to break your input and output operations into smaller functions. All other functions in this class will need the word 'static' in the header, such as the following:

```
public static int getValidInt(int min, int max) {
```

ConnectFour.java

First, you'll need to decide what instance varibles you'll need to represent a game of connect four. You'll need some variable to represent the state of the board, and a variable to represent whose turn it is.

Once you have those variables, you'll need to create methods that:

- Make a new board (this will be your constructor method)
- Allow a player to make a move based on their column selection.
- Detect a win, loss or draw
- Convert the board into a string

Note that no method call in your board should put the ConnectFour object in an invalid state. Even if someone wrote their own Project1.java, they should not be able to break your ConnectFour object. For the constructor, if someone passes invalid arguments, have the constructor set the board to be 5x5.

Grading

See the course website for a description of how your project will be graded.

Project Submission

Before submitting your project, be sure to compile and test your code on the GL system. See the Project Compilation section of the course Projects page for details on how to compile and execute your project on GL.

Assuming you've used the specified file names, submit your project by typing the command

```
submit cs202 proj1 Project1.java ConnectFour.java
```

See the Project Submission page for detailed instructions.

You may resubmit your files as often as you like, but only the last submittal will be graded and will be used to determine if your project is late. For more information, see the Projects page on the course web site.

Remember — if you make ***any*** change to your program, no matter how insignificant it may seem, you should recompile and retest your program before submitting it. Even the smallest typo can cause errors and a reduction in your grade.