Gebze Technical University
Department of Computer Engineering
CSE 241/505
Object Oriented Programming
Fall 2017
Homework # 5
Due date Nov 26th 2017

In this homework, you will write a hierarchy of **`ConnectFour`** classes. But first, there should be cell class (no inner class) a Connect Four cell. The class **`Cell`** will hold the position of the cell (A, B, C, etc) and the row number (1, 2, 3, etc). This class will also include all necessary setters/getters, constructors etc. Remember a cell can be empty, user (user1, user2) or computer.

Your base class **`ConnectFourAbstract`** will define the following member functions. It will be an abstract base class.

- The class will have functions to read and write from files. You will decide on the file format.
- The class will have functions to return the current width and height of the board
- The class will have a function that displays the current board on the screen
- The class will have two functions named play that plays the game for a single time step. First function does not take a parameter and it plays the computer. The second function takes a cell position and it plays the user.
- The class should have a function that returns if the game ended.
- The class should have a function named playGame. This function plays the game by asking the user the board size first then asks the user to play and the computer plays, etc.

The class **`ConnectFourPlus`** will derive from the abstract class. It will play very similar to regular Connect Four but it will only accept horizontally or vertically connected cells as the goal of the game.

Similarly, the class **`ConnectFourDiag`** will derive from the abstract class. It will play very similar to regular Connect Four but it will only accept diagonally connected cells as the goal of the game.

The class **`ConnectFourPlusUndo`** will be very similar to **`ConnectFourPlus`**. It can undo the moves of the user or the computer until the beginning of the game.

- You will dynamic allocation to keep your Cell data.
- You will separate the interface and the implementation of the classes. Since there are 5 classes, you will have at least 11 source files to be submitted if you include the driver code.
- You will use your own namespace to keep all the classes.
- Do not forget to indent your code and provide comments.
- Check the validity of the user input.
- **Test your programs very carefully at least with 2 different runs for each game type. Submit at least 3 saved files with your HW.**
- You should submit your work to the moodle page.