**COMP 1451 Lab 10-b ( 4 points )**

**Take-home lab**

For this lab you will write four classes:

**`GamePiece`** has a String field that indicates how it is displayed, e.g. "X".

GamePiece has a method with this signature:
`public boolean isLegalMove(Location a, Location b)`
A legal move is to either a higher-numbered row or to a higher-numbered column, or both. Any other move is not legal. (A GamePiece object does not know the dimensions of the board.)

**`Board`** has a two-dimensional grid (4 rows and 4 columns) of `GamePiece` objects.

`Board` has a method that populates the board so that there is a piece in every location.

`Board` has a method that displays the board contents. An empty location is indicated by "-".

`Board` has a method to move a game piece:
`public void movePiece(Location from, Location to) throws InvalidMoveException`

This method checks to see if the proposed move can be made. If either of the locations is outside the boundary of the board, the move cannot be made. If there is no piece in the "from" location the move cannot be made. If the game piece at the "from" location reports that the move is not legal, the move cannot be made. If the move cannot be made the method throws the custom checked exception. If the move can be made it moves the piece. Keep in mind that a future version of this game will have different kinds of pieces and design accordingly.

**`Location`** has an x-position and a y-position to indicate the coordinates of a `GamePiece` on a `Board`, with appropriate accessor methods.

**`InvalidMoveException`** is a custom checked exception to be thrown in any condition where a valid move cannot be made, with the relevant reason passed to the constructor when an exception object is created.

There is no input reader required for this lab, so to test your methods you must create Location objects to pass to the Board's movePiece() method. <u>You are not expected to catch the exception, just be sure it is thrown in the appropriate place. Catching the exception will be done in the play() method of the Game class that you will write for the final assignment.</u>

The take-home lab is due before the next class. Upload it to the appropriate D2L dropbox. You will demonstrate it to your instructor or TA during lab time but it must be already in the dropbox. A suggested solution will be discussed in class and labs not already in the dropbox will not receive any points.