BİL-211 Object Oriented Programming and Design
Project
Assigned: 13.06.2014
Due: Design report : 07.07.2014 11:59PM, Code : 19.07.2013 11:59PM
Submission: Compress all the files (.rar, .zip etc.) and send them to kaanbasal@etu.edu.tr by email.
Rules: <u>Late submissions are not allowed</u>. Plagiarism is strictly forbidden, university regulations will be applied for all that take part.

Chess is a two-player strategy board game. We require a program that allows two players play chess on a computer. You should display a chess board and take inputs from the two players one-by-one. The game has two modes: normal and time control. In time control mode, the players will have a certain amount of time to make all the moves; the first player that runs out of time loses the game, unless there was a checkmate before. In normal mode, the game continues until checkmate.

Regulations:

- All the standard chess rules apply. (see `http://en.wikipedia.org/wiki/Chess`)

- All types of pieces should be coded as separate classes that implement the following interface:

  ```
  public interface piece {
      //checks whether the piece can move from location (startx,starty) to (endx, endy)
      public boolean checkMove(int startx, int starty, int endx, int endy);

      //returns whether the piece is still on play or not.
      public boolean isCaptured();

      //return all moves that the piece can make. It should return an array of
      //coordinates (x,y), in a format designed by you.
      public int[] allMoves();
  }
  ```

- The movement of pieces have to be performed by using Strategy Pattern. You should move pieces by checking their strategy in a movement, such as (does not have to be exactly like this),

  ```
  .....
  //move a piece on (x0,y0) to (x1,y1) using p to determine the legality of the move.
  //p is of type piece.
  move(x0, y0, x1, y1, p);
  ....
  ```

  Therefore, somehow, you should be able to send the strategy of the piece to the movement control.

- The display of the board (the view) should be performed by using Observer Pattern.

- At the begining, the mode should be selected and the view in time control mode should also display a clock. That clock should run until the current user makes its move, then stop.

As your customer, we want you to write a program that provides these functionalities. All other details (the look of GUI, the classes, algorithms, I/O etc.) are all up to you. Please use piazza for other questions and comments.

Your project will be evaluated in two steps:

1. **Design Report :** A report that describes how you will implement the project. The items below should definitely be included. You can also add some other details/explanations that you think as necessary.

   - Use case scenarios, the main scenarios (4-5 is enough)
   - Detailed class diagram
   - Detailed sequence diagram (1-2 is enough)

2. **Demo :** At the end you are required to do a demo in which you will execute your program and tell us how it works. Demo schedule will be announced later. All members of the project group should participate and we expect that all members should answer the questions about any part of the project.