Homework 4 (30 points)

cs1007 - Object-oriented programming and design in Java
Prof. Shlomo Hershkop
Dept of Computer Science
Columbia University
Spring 2006

Hw Due: May 9 11pm (electronically)

This homework is geared towards fine tuning your programming and understanding of GUI/graphics/threading skills. It should not be too hard, if you finished the last homework in one piece…..please do not leave this to the last minute, start early and take advantage of our help…

You will be adopting a working (but not correct) game of ping pong in Java, porting it from Applet to Java Swing. You will also be practicing your thread programming skills.

I have made available the code online (see schedule page), it was featured in class on 4/18/06. Please download the code, and make sure you can get it running (at the very least).

To help you along, I am outlining step by step guidelines, so that if you complete each one in order, you are guaranteed maximum points on the homework.

1) Basic port
   a. First start by getting the applet version to run (ask for help if you need it here).
   b. You need to use threads in this project.
   c. Start early!!!
   d. Now look over the code, as we did in class, understand how the pieces work together
   e. Create a new Java project, you will be porting each of the classes from the example code to swing, so its better to code in empty class and fill in the details….feel free to rename or rethink the pieces of the program, make sure each of the next steps runs before continuing to the next step, MAKE SURE TO ADD COMMENTS TO YOUR CODE
      i. Don't do anything except get a blank window to open up (and close/exit correctly).
      ii. Port the field class so that you can display a colored field (switch the background color to whatever you like.
      iii. Port the ball class, so that you can get a bouncing ball on the screen. Don't forget to play around with the ball color.
      iv. Get the ball to bounce around the screen (for this step you can leave it as it is so that it bounces off the back wall back into the game
      v. Port the paddle so that it shows up in the game.
2) More complicated coding
      i. Add a score counter; so you can show the current score
      ii. Add a miss counter; to show how many times the paddle missed the ball (we will add it in later).
      iii. Add a hit counter; so that every time the ball is hit, the number is incremented….i.e. every time the ball and paddle meet, the number goes up by one.

    iv. Convert the ball class, it so that it works for the current window's size….and not as it is done…i.e. if I resize the window, it will still work.

    v. Make sure you are not hard coding any of your code to a specific window size

    vi. Change it so that if the ball misses the paddle (to the right of the paddle) it disappears and the miss counter is incremented.

    vii. Every 10 times you hit the ball you should get 1 point (on the score counter).

    viii. Add a JProgress bar (as shown in class) to show how close you are to a point.

3) Test to see if you can play a regular game of ping pong. If you spend more than 10 hours to get to this point, please make sure to ask for help (i.e. before you use up your 10 hours)……

4) Extra coding
  a. Add something snazzy to the game, you can choose to implement only one of the following:

    i. Every time you get a point (i.e. every 10 bounces) add a second ball to the game. But as long as a ball exists the game continues…and makes sure the second ball isn't exactly where the first is….you should make it appear on the far left when the score is generated so that the player has a fair chance.

    ii. Add a magic square somewhere on the left side of the board….it needs to be hit twice to get destroyed and give the player something special.

    iii. Allow the up and down arrows on the keyboard to move the paddle, and the space bar to add a new ball every time it is pressed ☺

5) Extra Credit:
  a. Add something original to the game to make it fun, contact me if you are not sure what you want to do will qualify.