CS 201: Final Exam

Do not discuss this exam with anyone until after 4pm Monday 12/16!

For full credit, all answers should be clear and concise.

Problem 0 (4 points): Write your name at the top of this page.

Problem 1 (10 points): For each of the following values, write the declaration of a variable to store that value. Be sure to choose an appropriate type and variable name.

(a) The number of undergraduate students at Loyola College.

(b) The student to faculty ratio at Loyola College.

(c) The average distance, in miles, from the Sun to Pluto (which is about 40 times greater than the 93 million miles from the Sun to Earth).

(d) The value of π to 10 decimal places.

(e) An index into an array of `doubles`.

Problem 2 (10 points): Write a method so that the following code fragment

```
g.setColor(Color.black);
g.drawLine(100, 150, 200, 150);
g.drawLine(200, 150, 150, 120);
g.drawLine(150, 120, 100, 150);

g.setColor(new Color(100, 50, 200));
g.drawLine(20, 40, 30, 40);
g.drawLine(30, 40, 25, 20);
g.drawLine(25, 20, 20, 40);
```

can be replaced with the following method calls

```
drawTriangle(g, 100, 150, 100, 30, Color.black);
drawTriangle(g, 20, 40, 10, 20, new Color(100, 50, 200));
```

Problem 3 (16 points):

(a) Recall that the `Card` class has two instance variables `rank` and `suit`, both `int`s. Write a method called `pointValue()` that determines the value the card the method is invoked on would have in the game of Hearts. `pointValue` should return 1 if the card is a heart, 13 if the card is the queen of spades, -10 if the card is the jack of diamonds, and 0 otherwise. See below for an example of how `pointValue` would be used.

```
Card a = new Card(3, Card.HEARTS);
Card b = new Card(4, Card.CLUBS);
Card c = new Card(Card.JACK, Card.DIAMONDS);
Card d = new Card(Card.QUEEN, Card.SPADES);

System.out.println(a.pointValue()); // should output 1
System.out.println(b.pointValue()); // should output 0
System.out.println(c.pointValue()); // should output -10
System.out.println(d.pointValue()); // should output 13
```

(b) Complete the following method `trickValue` that, given an array of four cards, computes and returns the sum of their values as determined by the `pointValue` method. You may assume that you have a working version of `pointValue`.

```
public static int trickValue(Card[] cards)
```

Problem 4 (12 points): The following method is intended to return the area, in hectares, of a rectangular plot of land given, as parameters, the coordinates (in meters) of opposite corners of the plot. As written, it does not accomplish its task. (Note: there are 10000 square meters in a hectare).

```
public double computeArea(int x1, int y1, int x2, int y2)
{

  int width = x2 - x1;


  int height = y2 - y1;


  if (width < 0)
  {


    width = -width;


  }
  else if (height < 0)
  {


    height = -height;

  }


  return width / 10000 * height;


}
```

(a) `computeArea(0, 0, 100, 50)` should return `0.5`. What does it return?

(b) `compute Area(10000, 10000, 0, 0)` should return `10000`. What does it return?

(c) Make corrections to the method so it works properly. Also, add the definiton of an appropriate constant and modify the method so it uses that constant.

Problem 5 (12 points): The following method was purposefully formatted poorly and has errors.

```
public int foo(int x, int y)
{


  if (x == 1 || y > 0)


  if (y < x)
  {


  if (x > 0)


    return 1;


  }
  else


    return 0;


}
```

(a) What is the return value of `foo(1, 1)`?

(b) Find a value of `x` and `y` for which the method returns a value different than your answer to (a).

(c) Find a value of `x` and `y` for which the method does not reach a `return` statement.

(d) Add one statement (any suitable statement will do) that will resolve the compiler error.

Problem 6 (16 points): Write statements equivalent to the following. Use loops so that each answer contains only one statement that invokes `twiddle` (that statement may be executed many times).

(a)
```
obj.twiddle(1);
obj.twiddle(3);
obj.twiddle(5);
obj.twiddle(7);
```

(b)
```
obj.twiddle(arr[0]);
obj.twiddle(arr[1] + 5);
obj.twiddle(arr[1] + 10);
obj.twiddle(arr[2] + 10);
obj.twiddle(arr[2] + 15);
obj.twiddle(arr[2] + 20);
```

Problem 7 (10 points): Add the code necessary to complete the given applet so that when the user clicks the button, the number displayed in the `TextField` increases by 1. Note that none of the variables have been declared. Add the declarations in such a way that only the variables that absolutely must be instance variables are declared as instance variables.

```
import java.applet.*;


import java.awt.*;


public class Question7 extends Applet
{


    public void init()
    {


      value = 0;


      button = new Button(``Increment'');


      add(button);


      output = new TextField(String.valueOf(value));


      add(output);


    }


    public void actionPerformed(ActionEvent e)
    {


      value++;


    }
}
```

Problem 8 (10 points): Write a new method in the `ChessBoard` class called `totalWhitePieces` that adds up the values of the white pieces on the board and returns that total. Assume that there is a method `int pieceValue(char piece)` that returns the value of the piece represented by the character passed to it (so for 'p' and 'P' it returns 1 since pawns are worth 1 point, for 'h' or 'H' it returns 3, and so on).

Some of the declarations of methods in the `ChessBoard` class are given below to refresh your memory.

```
public class ChessBoard
{
    private char[][] pieces; // holds letter codes for pieces (' ' for empty)

    public static final int SIZE = 8;
    public static final int WHITE = 0;
    public static final int BLACK = 1;

    // determines if the given square is empty
    public boolean isEmpty(int r, int c) { // code omitted }

    // returns the color (BLACK or WHITE) of the piece at the given location
    // return value is meaningless if there's no piece there
    public int getPieceColor(int r, int c) { // code omitted }

    // returns the value of the piece represented by the given character
    // return value is meaningless if the character does not represent a piece
    public static int pieceValue(char piece) { // code omitted }

    // other methods and constructor omitted

    // your method would go here
}
```