Midterm I - CSE11 – Fall 2013
CLOSED BOOK, CLOSED NOTES
50 minutes, 100 points Total.

Name: _____ ID: _____

Problem 1) (8 points)
For the following code segment, what are the values of i, j, k, and d, after the segment executes?

```
int j = 70;
int i = 20;
int k = j/i;
double d = 15;

j += i;
k *= 10;
d = (d + j)/i;
j -= i++;
```

i = 21 (i++)
j = 70 (add i, subtract i)
k = 30 (integer division, then mult)
d =5.25 (integers promoted to double
(15.0 + 90)/20 = 105.0/20)

Problem 2 (8 points) Circle the letter next to each valid java identifier.

a. `f`
b. `forget_me_now`
c. `2ndStreet`
d. `Script$$`
e. `To-be`
f. `under_water`
g. `Elmo's`
h. `ThisIsSportsCenterOnESPN`

Problem 3) (10 points) Read the following code segment. What does it print out?

```
int i = 5, sum = 0, j;
while (i > 0 ) {
     j = 1;
     while (j <= i ) {
          sum++;
          j++;
     }
     System.out.println("i = " + i + " sum = " + sum);
     i -= 1;
}
```

Answer:
i = 5 sum = 5
i = 4 sum = 9
i = 3 sum = 12
i = 2 sum = 14
i = 1 sum = 15

Problem 4 (12 points)

Find all syntax errors in the following code. Circle each error and label with a number (e.g. 1)

Following the code segment on the lines provided, Describe briefly in English the actual syntax error the corresponds to the number by the circled error.

```
public class Midterm1
     private static final int MAXCOUNT = 85
     public static void main(String[] args)
     {
          int count = 0.0;
          if ( count > MAXCOUNT )
          {
               System.out.println('Count exceeded”];
          }
          return count;
     }
}
```

1. ___Missing opening brace_____

2. ____Missing Semicolon _____

3. ____literal is double, it should be an int _____

4. ____String literal should be in double quotes _____

5. ____Method invocation should use ')' not ']' _____

6. ____void method cannot return a value _____

Problem #5 (16 points)
Write a public method called `addOdd` that takes a single integer argument n. `addOdd` should return an integer that is the sum of odd numbers in the interval [1,n]. Use a `while` loop in this method to perform the computation. If n is a negative number, `addOdd` should return 0. Only code the method, do not write a class definition.

ans:

```
public int addOdd(int n)
{
     if (n < 0)              // this test is optional
          return 0;
     int sum = 0;            // sum of odd numbers
     int i = 1;
     while (i <= n)          // 1 .. n, inclusive
     {
          sum += i;
          i += 2;            // step through odd numbers
     }
     return sum;             // return the computed sum
}
```

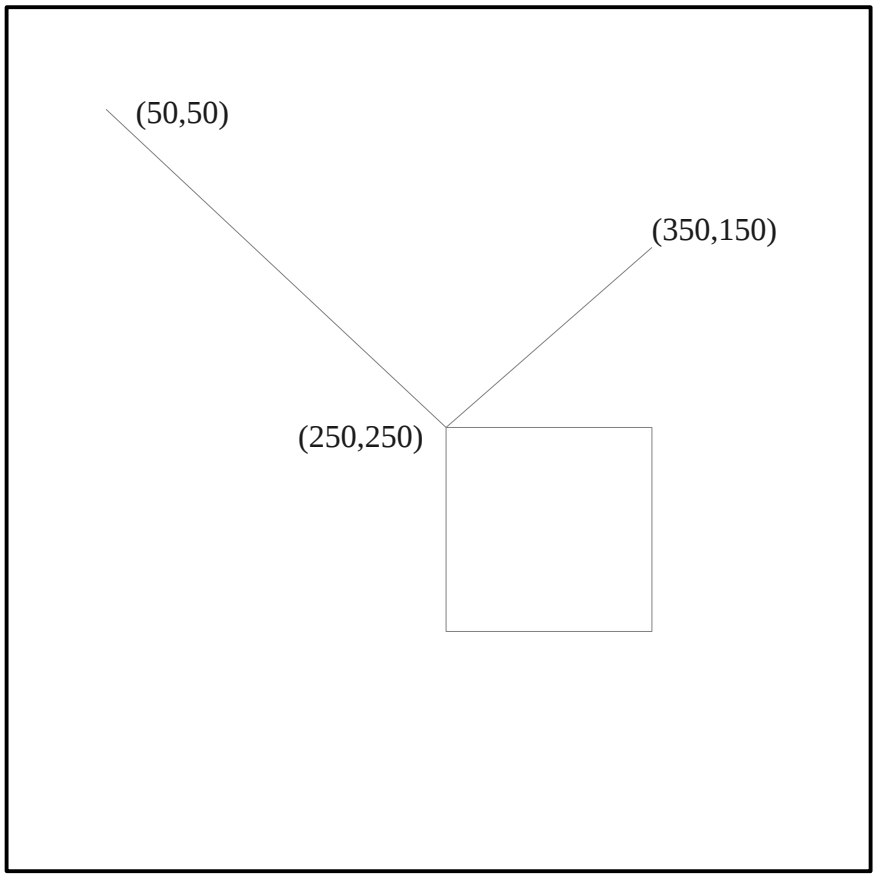
Problem #6 (8 points) Assume x,y,z are integers with the values, x = 17, y=11, z=3 what are the values of the following expressions

```
(a) (x % 5) + 1 == z
(b) x > y && z * 4 < y
(c) 11 + 2*z == x++
(d) z < y || x/2 + z – 14 + y % 4 > 11
```

ans:
(a) (17 % 5) + 1 = 2 + 1 = 3 3 ==3 ==> **True**
(b) (17 > 11) && (12 < 11) ==> True && False ==> **False**
(c) (11 + 6) == 17 ==> **True** (x is post-incremented)
(d) (3 < 11) || (??) ==> **True** (short circuit the OR, no need to evaluate right hand expression)

Problem #7 (12 points) Read the following code that uses methods defined in objectdraw (used by the textbook and in your programming assignments). Assume the framed box below is a 500x500 drawingCanvas. Draw what should be on the canvas, after the code executes. For each `Line` object, clearly label the coordinate of the line endpoints. Freehand drawing is acceptable (rulers not required)

```
import objectdraw.*;
import java.awt.*;
public class DrawIt extends WindowController {
    private static final int WIDTH=500;
    private static final int LX=250;
    private static final int LY=250;
    private static final int EDGE=100;
    public void begin()
    {
          Location origin = new Location(50,50);
          Location thePoint = new Location(LX,LY);
          new FramedRect(LX,LY,EDGE,EDGE,canvas);
          new Line(origin, thePoint, canvas);
          new Line(LX,LY,LX+EDGE,LY-EDGE,canvas);
    }
    public static void main(String[] args) {
          new DrawIt().startController(WIDTH,WIDTH);
    }
}
```



Problem #8). (20 points) Circle T (indicating True) or F (indicating False) for each one of the following.

T	F	Only static variables are initialized to zero (or Null , or the empty string) by java
T	F	`void mu(Epsilon x);` and `void mu(Epsilon y);` have identical signatures
T	F	Constants must be declared private
T	F	A static class variable is shared by all instances in the class
T	F	It is legal to use the `public` access modifier to declare a temporary variable within the statement block that defines a method
T	F	If the file called `yourClass.java` defines a public class, the class name must be named your`Class`
T	F	It is legal to perform an assignment within a boolean expression
T	F	Variables of type `double` are automatically converted to type `int` in arithmetic expressions
T	F	Variables of type `int, float, double, bool,` and `char` are considered primitive types
T	F	A class can only define one constructor
T	F	The java compiler used in your homework is called `javac`
T	F	Java applets have a `main` method defined, while java programs do not
T	F	`%` is called the mod operator and computes the remainder when performing division of two integers
T	F	`while` loops always terminate
T	F	The `onMouseClick` method used in homework programs is called an event handler
T	F	`double d = 1.0; d++;` is a valid sequence of two java statements
T	F	`If` statements must always have an `else` clause
T	F	`( x == y || i < 25 )` is equivalent to `!(x != y && i >= 25)`
T	F	To access the x coordinate of a `Location` instance called `loc`, the proper expression is `loc.getX()`
T	F	The unix command "`ls .`" will list the contents of the current directory.

Problem #9 (6 points) What will the following code segment print out? Why?

```
import java.awt.*;
import objectdraw.*;
public class Testing {
      public static void main(String[] args)
      {
            Location coord1, coord2;
            int x = 2, y = 2;

            coord1 = new Location(x,y);
            coord2 = new Location(x,y);
            if (coord1 == coord2 ) {
                  System.out.println("Same Coordinates");
            }

            if ( x == y ) {
                  System.out.println("Same Coordinate Points");
            }
      }
}
```

ans:

it prints out "Same Coordinate Points"

even though coord1 and coord2 are constructed with identical parameters, they are different objects.
The first if statement compares if the variables refer to the same object and is false.
The second if statement tests if the values of x and y are the same. This is true.