Midterm Exam I

CS 312 Introduction to Programming

24 Feb 2015

Name: _____

EID: _____

Problem Number	Topic	Points Possible	Points Off
1	Expressions	18	
2	Code Tracing	18	
3	Syntax Errors	10	
4	Method Tracing and Parameters	10	
5	Programming and Loops	20	
6	Programming Using Graphics	25	
7	Programming	15	
	Total points off		
	Score out of 116		

Instructions:

1. Please turn off your cell phones
2. There are 7 questions on this test.
3. You have 2 hours to complete the test.
4. Place your answers on this test, not scratch paper.
5. You may not use a calculator.
6. When code is required, write Java code. You may use only features that we discussed up to topics 1-12, including those covered in the textbook for that material (Chapters 1-4).
7. Style is not evaluated when grading.
8. If you believe a question has an error or is ambiguous, state your assumption and answer based on your assumption.
9. If you finish early bring your exam and scratch paper to the proctor and show them your UTEID.

Name: _____

EID: _____

1. Expressions (1 point each, 18 points total). For each Java expression in the left hand column, indicate the result of the expression in the right hand column. *You must show a value of the appropriate type. For example, 7.0 rather than 7 for a double and "7" instead of 7 for a String. Answers that do not indicate the data type correctly are wrong.*

A. `8 + 5 * 3 / 2` _____

B. `3 * (7 + 1) - 3 - 7 + 2` _____

C. `23 % 5 + 31 / 4 % 3 - 17 % (16 % 10)` _____

D. `"1" + 2 + 3 + "4" + 5 * 6 + "7" + (8 + 9)` _____

E. `1 + 1 + "(8 - 2)" + (8 - 2) + 1 + 1` _____

F. `29 / 4 / 2.0 + 18 / 5 + 1.5` _____

G. `5 / 2 + 123 / 10 / 10.0` _____

H. `13 % 5 + 43 % (11 % 3)` _____

I. `-(6 + 3 - 2 * 3)` _____

J. `9 / 2 / 2.0 + 9 / 2.0 / 2` _____

K. `1 + "x" + 11 / 10 + " is" + 10 / 2` _____

L. `1 / 2 + -(157 / 10 / 10.0) + 9.0 * 1 / 2` _____

M. `2 + "(int)2.0" + 2 * 2 + 2` _____

The `Math` methods `ceil`, `floor`, `sqrt`, `pow`, and `abs` all return doubles.

N. `Math.round(-6.4) + Math.ceil(6.5)` _____

O. `Math.floor(Math.max(Math.min(-5, 5.5), Math.max(-4.5, -6)))` _____

P. `Math.pow(3, 2)` _____

Q. `Math.ceil(5.2) + Math.floor(-5.2)` _____

R. `(double) 3 / 4 + (int) (5.0 / 6 + 1 / 6.0)` _____

Name: _____

EID: _____

2. Code tracing (2 points each, 18 points total). Place you answer in the box to the right of the code. If the code results in a compiler or runtime error, state the kind of error that occurs.

A. What is output by the following code when it is run?

```
int xa = 2;
xa = -3;
xa -= -5;
xa++;
System.out.print(xa);
```

B. What is output by the following code when it is run?

```
int xb = 4;
xb++;
int zb = xb + 2;
double ab = zb / xb;
ab -= 2 + xb;
System.out.print(xb + " " + zb + " " + ab);
```

C. What is output by the following code when it is run?

```
int xc = 3;
int yc = 4;
String sc = "xc";
xc++;
sc = xc + yc + sc;
System.out.print(xc + " " + sc);
```

D. What is output by the following code when it is run?

```
int yd = 2;
for(int j = 0; j <= 3; j++) {
  int xd = 3 - j;
  yd = yd + xd;
}
System.out.print(yd);
```

E. What is output by the following code when it is run?

```
int ae = 5;
int be = 12;
double xe = be / ae;
int re = be % ae;
System.out.print(re + " " + xe);
```

Name: _____

EID: _____

F. How many asterisks does the following code print out? (Don't show the output. Simply state the number of asterisks that are printed out when the code runs.)

```
for (int i = 1; i <= 5; i++) {
  for (int j = 0; j < 5; j++) {
    System.out.print("*");
  }
}
```

G. How many asterisks does the following code print out? (Don't show the output. Simply state the number of asterisks that are printed out when the code runs.)

```
for (int i = 1; i <= 10; i++) {
  for (int j = i; j <= 10; j++) {
    System.out.print("*");
  }
}
```

H. How many asterisks does the following code print out? (Don't show the output. Simply state the number of asterisks that are printed out when the code runs.)

```
for (int i = 7; i > -3; i--) {
  System.out.print("*");
  for (int j = 0; j < i; j++) {
    System.out.print("*");
  }
  System.out.println("*");
}
```

I. What is output by the following code when it is run?

```
int x = 5;
int y = 13;
if (x % y == 3) {
  System.out.print("A");
} else if (x + y > 15) {
  System.out.print("B");
}
if (x >= 5) {
  System.out.print("C");
} else {
  System.out.print("D");
}
```

Name: _____

EID: _____

3. Syntax errors (10 points). Each of the following code snippets contains a syntax error. Explain what the syntax error is in a single sentence.

A.

```
int aa = 12;
int i = 31;
double da = i * aa++ - 2;
for (int i = 0; i < da; i++) {
  System.out.print("*");
}
// What causes the syntax error?
```

B.

```
int _bx;
_bx = 5;
double bz = 3;
int by = Math.pow(_bx, bz);
System.out.print(bz + " " + by * bz);
// What causes the syntax error?
```

C.

```
int cx = 5;
double 2x = 2 * cx;
String cs = "CS" + 2x;
System.out.print(cx + " " + cs);
// What causes the syntax error?
```

D.

```
double sum;
int dz = 5;
for (int i = 1; i <= dz; i++) {
  sum /= i;
}
System.out.println(sum);
// What causes the syntax error?
```

E.

```
int ea = -3;
double ez = 5 * ea;
if (ez = 5) {
  System.out.print(ez);
}
// What causes the syntax error?
```

Name: _____

EID: _____

4. Method Tracing and Parameters (10 points): At the bottom of the page, write the output produced by the following program, as it would appear on the console.

```
public class ParameterMystery {
    public static void main(String[] args) {
        int one = 4;
        int two = 3;
        int three = 10;
        int num = 17;
        int four = 3;

        racket(one, two, three);
        racket(three, four, 5);
        racket(2, two * 2, num);
        racket(num, three * one, four);
        racket(three - four, one, two + 8);
    }

    public static void racket(int two, int one, int three) {
        System.out.print(three + " is roughly " + two + " plus " + one);
        if (one + two > three) {
            bark();
        } else if (one >= three - two) {
            sneeze();
        } else {
            holler();
        }
        System.out.println();
    }

    public static void holler() {
        System.out.print(" YES");
        bark();
        sneeze();
    }
    public static void bark() {
        sneeze();
        System.out.print(" NO");
    }
    public static void sneeze() {
        System.out.print(" maybe");
    }
}
```

Name: _____

EID: _____

5. Programing and Loops (20 points).

Write a Java method to produce the following output. The method relies on a parameter named `size`.

Following is the output for various values of `size`:

output with `size` 3:	output with `size` 5:	output with `size` 6:

`***`
`**-|`
`*-|-`
	`*****`
	`****-|`
	`***-|-`
	`**-|--|`
	`*-|--|-`
		`******`
		`*****-|`
		`****-|-`
		`***-|--|`
		`**-|--|-`
		`*-|--|--|`

Complete your method, including the method header, in the space provided:

Name: _____

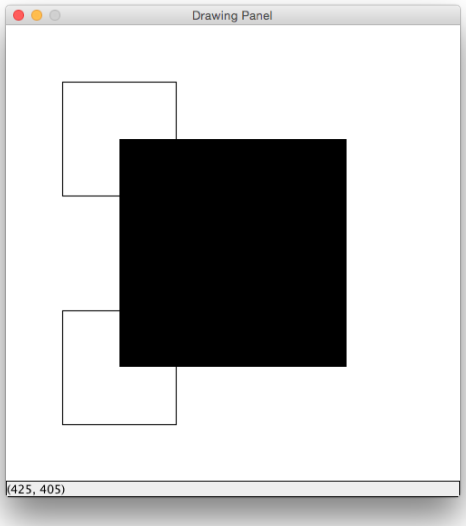
EID: _____

6. Programming using Graphics (25 points). Complete a method to produce the following output. The parameters for the method are the `Graphics` object for the `DrawingPanel`, the size of the `DrawingPanel`, and the number of squares to produce. Assume the `DrawingPanel` is square with length and width equal to the `size` parameter.

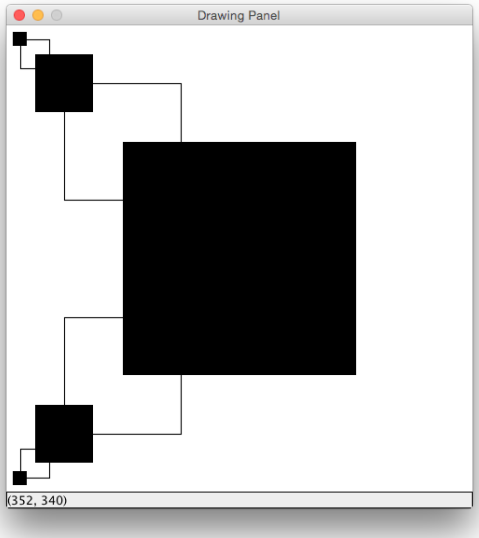
Your method must be general and work for various values of the size of `DrawingPanel` and various numbers of squares.

Assume the color of the `Graphics` object has already been set to `Color.BLACK`.

Here is the output of the method call `drawInfinity(g, 500, 2);`



Here is the output of the method call `drawInfinity(g, 500, 5);`



Complete the method on the following page.

Name: _____

EID: _____

(*Continued from previous page.*) Complete the following method:

```
// Assume size is the drawing panel width and height
public static void drawInfinity(Graphics g, int size, int numSquares) {
```

Name: _____

EID: _____

7. Programming (15 points)

A recent study showed a link between college GPA and starting salary at your first job (ask me about it in class if you're interested). Write a method named `startingSalary` that will return the amount you should expect upon graduation. The starting salary is determined by your GPA, the total number of credits you have earned, and the number of honors credits you have earned, in the following fashion:

- Students must have completed at least 180 credits and have a GPA of at least 2.0 to graduate. Any student with less than these amounts will not graduate, and will receive a $0 starting salary.
- For students who have sufficient credits to graduate and a sufficiently high GPA, the starting salary will be determined by their GPA:
    - A *normal* salary for this position is $77,700
    - A *low* salary will be given to students with a GPA of 2.8 or less, and is $65,000
    - A *high* salary will be given to students with a GPA of 3.8 or more, and is $115,700.
- Honors credits can boost a student's GPA. Students who earned at least 15% honors credits will receive a 15% boost in their GPA. Note: a student's boosted GPA will be calculated *before* it is classified as *normal*, *low*, or *high*, but no amount of honors credit can help you graduate (i.e. if your raw GPA is less than 2.0, you will still not graduate).

Here are some example calls to the method and their resulting return values:

Call	**Returning Value**
`startingSalary(3.87, 178, 16)`	`0`
`startingSalary(1.99, 185, 55)`	`0`
`startingSalary(2.7, 380, 50)`	`65000`
`startingSalary(3.7, 200, 29)`	`77700`
`startingSalary(3.7, 200, 30)`	`115700`
`startingSalary(3.8, 185, 0)`	`115700`

You may assume that the GPA will be between 0.0 and 4.0 and that both credit counts will be non-negative integers.

Provide the method, including the method signature, on the following page.

Name: _____

EID: _____

Name: _____

EID: _____

```
// Complete the answer to problem 7 here
```