

Name: _____

CS 141: Introduction to (Java) Programming: Exam 1*Jenny Orr • Willamette University • Fall 2015*

1.	(max 12)	4.	(max 20)
2.	(max 14)	5.	(max 20)
3.	(max 14)	6.	(max 20)
Total:		(max 100)	

1. (3 pts each, 12 pts total) What is the value of the variable **answer** after each of the following statements. Assume the declarations below.

```
double answer;
int n = 5;
int m = 8;
```

a. `answer = n % m;` 5 % 8 = 5 _____

b. `answer = m % n;` 8 % 5 = 3 _____

c. `answer = 4*(n/m);` 4 * (5/8) = 0 _____

d. `answer = 1-3.0/2+10;` 1 - 1.5 + 10 = 9.5 _____

2. (1 pts each, 14 pts total) **True and False:** Please circle T or F
(Credit is only given if the instructor can clearly tell which answer is circled)
- a. **T or F:** In general, member variables should be public.
 - b. **T or F:** The name of a constructor must be the name of the class.
 - c. **T or F:** A constructor should always have a void return value.
 - d. **T or F:** The process of hiding object data and providing methods for data access is called encapsulation.
 - e. **T or F:** An object's accessor method is called when the keyword `new` is used.
 - f. **T or F:** An object's *member* variable exists for as long as the object exists.
 - g. **T or F:** *Private methods* can be called outside of the class by using setters and getters.
 - h. **T or F:** The declaration:

```
Card c;
```

creates a new Card object.
 - i. **T or F:** It is never ok for two different variables within a class to have the same name.
 - j. **T or F:** Methods can have multiple arguments and can return multiple items on a single return.
 - k. **T or F:** Java source files end with the `.class` extension.
 - l. **T or F:** The value of a char variable can only be a letter of the alphabet.
 - m. **T or F:** In an assignment statement, the value of the expression on the left of the equals-sign is copied into the variable on the right.
 - n. **T or F:** A relational operator compares two Boolean expressions.

3. (14 pts total) Boolean Expressions

a. (2 pts) DeMorgan's Laws: Given the declarations:

`int x;``int y;`The Boolean expression: `!((x < 10) && (y < 20))`

is identical to which of the following (circle correct answer)

a) `(x < 10) || (y < 20)`b) `!(x < 10) && !(y < 20)`c) `(x >= 10) && (y >= 20)`d) `!(x < 10) || !(y < 20)`

b. (12 pts) Boolean Expressions: Given the declarations:

`int z;``char c;``String w1, w2;`

Write a Boolean expression for the following:

a. The mathematical expression $50 < z < 200$ 50 < z && z < 200

b. The character c is either the letter A or the letter Z.

c == 'A' || c == 'Z'

c. Words stored in variables w1 and w2 are the same.

w1.equals(w2)

4. Scope and Execution (20 pts total) For each variable listed at the top, indicate a "D" on the line where the variable is declared. Mark an "x" to indicate scope.

a) Scope (14 pts)

		args	st	x	i	b	a	b	n	i
1	public class MyScopeProject {									
2	public static void main(String[] args)	D								
3	{									
4	ScopeTest st = new ScopeTest();	x	D							
5	}	x	x							
6	}									
7	public class ScopeTest									
8	{									
9	private int x = 2;			D						
10	public ScopeTest() {			x						
11	int i = 10;			x	D					
12	int b = g(i);			x	x	D				
13	System.out.println(b + i);			x	x	x				
14	}			x	x	x				
15	public int g(int a){			x			D			
16	int b = 0;			x			x	D		
17	for (int n = 0; n < a; n++) {			x			x	x	D	
18	int i = x*n;			x			x	x	x	D
19	b = b + i;			x			x	x	x	x
20	}			x			x	x	x	x
21	return b;			x			x	x		
22	}			x						

b) (6 pts) Execution order: The program begins executing at Line 2. What are the next 6 lines that are executed?

Execution Step 0: Line 2

Other acceptable answer: 2,3,8,9,10,11,14

Execution Step 1: 3

Execution Step 2: 9

A few other variations may also be accepted.

Execution Step 3: 10

Execution Step 4: 11

Execution Step 5: 14

Execution Step 6: 15

5. (10 pts each, 20 pts total) Loops

- a. Write a while-loop which prints out the numbers from 0 to 50 (inclusive).

```
public class LoopPractice {
    public static void main(String[] args) {

        int i = 0;
        while (i <=50) {
            System.out.println("i = " + i);
            i++;
        }
    }
}
```

- b. Write a for-loop which generates 100 random numbers (use Math.random()), counts up how many of them are less than 0.5, and prints the count at the very end. If you have trouble getting started, begin by at least generating 100 random numbers, then think of how to count up the number that are less than .5.

```
public class LoopPractice {
    public static void main(String[] args) {

        int cnt = 0;
        for (int i = 0; i < 100; i++) {
            double r = Math.random();
            if (r < .5) cnt++;
        }
        System.out.println("The number less than .5 is " + cnt);

    }
}
```

6. (20 pts) Create a `Car` class consisting of the following:
- Two member variables: one for the car's make (e.g. Honda) and one for its mpg (e.g. 52.5) (*make sure you use the appropriate data type such as `String`, `char`, `int`, `double`, etc).*)
 - A constructor which has 2 parameters used for setting the value of each member variable.
 - A setter & getter for the mpg variable.
 - A `toString` method.

```
public class Car {  
  
    // Member variables for model and mpg:  
    private String make;  
    private double mpg;  
  
    // Constructor:  
    public Car(String mk, double mp) {  
        make = mk;  
        mpg = mp;  
    }  
  
    // Getter and Setter for mpg  
  
    public double getMpg() {  
        return mpg;  
    }  
  
    public void setMpg(double m) {  
        mpg = m;  
    }  
  
    // toString  
  
    public String toString() {  
        return "car make = " + make + " mpg = " + mpg;  
    }  
  
}
```