

**Fall 2017 CMP167 Final Exam V1**

**Name:** \_\_\_\_\_

**Date:** \_\_\_\_\_

1)	Use the code below to show the output for question 1a, 1b, 1c.	Method Call	Output
	<pre>public static void magicDiv(int x, int y){     int z = x/y;     System.out.println("z= "+ z); }</pre>	1a) magicDiv(10,3);	
		1b) magicDiv(10,2);	
1c) magicDiv(10,0);			

2)	Use the code below to show the output for 2a, 2b, 2c.	Method Call	Output
	<pre>public static void funPrint(int a){     if(a % 3 ==0){         System.out.println("a="+a);         for(int i=0; i&lt;a; i=i+2){             System.out.println("i="+i);         }     }     else{         for(int i=a; i&gt;0; i--){             System.out.println("zap "+i);         }         System.out.println("a="+a);     } }</pre>	2a) funPrint(4);	
		2b) funPrint(6);	
2c) funPrint(0);			

3)	Use the code below to show the result for question 3. If there is an error, write Error and what went wrong.	Write the content of the arrays
	<pre>int [] arrA = {7, 96, 13, 22, 15}; int [] arrB = new int [arrA.length]; int [] arrC = arrA; arrB[2] = arrA[2]; arrB[4] = arrA[0]; arrA[3] = arrC[1];</pre>	arrA -->
		arrB -->
arrC -->		

4) Write a **public static** method named **getAverage** that takes in an **array of int values** and **returns the average** as a double of all those values. Do NOT print the average.

5) Write a **public static void** method named **happyNewYear** that takes in an int and does the following:

- prints each number as it counts down to zero
- If the number is odd, print the current number followed by "OUU!"
- If the number is even, print the current number followed by "AAA!"
- When it reaches Zero, print "HAPPY NEW YEAR!"

6) Complete a **public class** to represent a **House** as described below.

a. Create the following private member variables

- int numWindows
- String ownerName
- boolean hasStairs

b. Create the default constructor.

c. Create the overloaded constructor that accepts all variables as input parameters.

d. Create **getter** methods for the variables.

e. Create **setter** methods for the variables.

f. Create the **equals(Object o)** method for the House object, such that it returns true if the values of all the members of both the calling object and the passed in object match. Return false if any values do not match.

g. Create the **toString()** method for the House object such that it returns a well structured sentence containing the information for all the object's variables.

Example of the String returned:

"House: Number of windows = 8, Owner = Hulk, has stairs.

"House: Number of windows = 6, Owner = Spiderman, does not have stairs.

h. Create 2 instances of House objects using your overloaded constructors from 6c with the following values.

Example of the values for h1, and h2:

h1 ---->

numWindows	8
ownerName	"Hulk"
hasStairs	true

h2 ---->

numWindows	6
ownerName	"Spiderman"
hasStairs	false

//This page was intentionally left blank.