**44-542 Object Oriented Programming Exam 01 Part 02 (55 Points) KEY**

1. (3 pts) Find the output of the following code segment:

**System.out.print("Carmen");**

**OUTPUT**

**CarmenCarmenCarmen**

**CarmenCarmen**

**System.out.print("Carmen");**

**System.out.println("Carmen");**

**System.out.print("Carmen");**

**System.out.print("Carmen");**

1. (4 pts) Suppose we have the following class:

**public class Movie {**

**private String title;**

**private String dirName;**

**private int length;**

**private boolean awardWinner;**

**public Movie(String movieTitle) {**

**title = movieTitle;**

**}**

**@Override**

**public String toString() {**

**return title + " " + dirName + " " + length + " " +**

**awardWinner;**

**}**

**}**

Suppose we now write a driver program containing the code shown below. Find the output.

**Movie myMovie = new Movie("Lord of the Rings");**

**System.out.println(myMovie);**

**OUTPUT**

**Lord of the Rings null 0 false**

1. (6 pts)Find the output of the following code segment:

**OUTPUT**

**12 12.0 12.5**

**int a = 50;**

**int b = 4;**

**int c = a / b;**

**double x = a / b;**

**double y = ((double) a) / b;**

**System.out.println(c + " " + x + " " + y);**

1. (4 pts)Find the output of the following code segment:

**OUTPUT**

**true**

**true**

**false**

**true**

**String str1 = "Leonard";**

**String str2 = "Leonard";**

**String str3 = new String("Leonard");**

**System.out.println(str1 == str2);**

**System.out.println(str1.equals(str2));**

**System.out.println(str1 == str3);**

**System.out.println(str1.equals(str3));**

1. (3 pts) Write the method header (that is, the first line of the method) for a method named funnyStuff. It is of type public and returns a String value. It has two parameters, the first of type int named myInt, and the second of type boolean and named myBoolean.

**public String funnyStuff(int myInt, boolean myBoolean)**

1. (4 pts)Suppose we have the following **Movie** class. Read the comments preceding the constructor and add the missing code in the space provided. Do not change any existing code.

**public class Movie {**

**private String title;**

**private String dirName;**

**private int length;**

**private boolean awardWinner;**

**/\*\***

**\* Assigns the values stored in the parameters to the**

**\* corresponding attributes.**

**\* @param title The title of the movie**

**\* @param dirName The name of the director of the movie**

**\* @param length The length of the movie in minutes**

**\* @param awardWinner A boolean value that tells whether the**

**\* movie is an award winner (true) or not (false)**

**\*/**

**public Movie(String title, String dirName, int length,**

**boolean awardWinner) {**

**// ADD MISSING CODE HERE**

**this.title = title;**

**this.dirName = dirName;**

**this.length = length;**

**this.awardWinner = awardWinner;**

**}**

**}**

1. (5 pts)Find the output of the following code segment:

**int x = 5;**

**OUTPUT**

**6 9 6**

**6 9 7**

**6 9 8**

**6 9 9**

**WOW!**

**7 8 7**

**7 8 8**

**WOW!**

**WOW!**

**int y = 10;**

**while (x < y) {**

**x++;**

**y--;**

**for(int z = x; z <= y; z++) {**

**System.out.println(**

**x + " " + y + " " + z);**

**}**

**System.out.println("WOW!");**

**}**

1. (6 pts)Trace the following code segment. Your trace must show all values assigned to the variables a, b, and c.

**TRACE**

**a b c**

**20 0 0**

**14 5 1**

**14 10 2**

**15 0**

**9 1**

**9 2**

**10**

**Values in blue are optional**

**int a = 20;**

**for(int b = 0; b < a; b += 5) {**

**a -= 6;**

**for(int c = 0; c < 2; c++) {**

**a += c;**

**}**

**}**

1. (5 pts)Suppose we have a person's name stored in a variable **personName** of type **String**; the person's age in a variable named **age** of type **int**; the person's monthly pay in a variable named **monthlyPay** of type **double**. We want to use the **printf** method to print this information formatted as shown here:

**Alistair 35 years old $ 3055.40 monthly pay**

The name of the person is left justified in a field of width 10, followed by a single space. The age is right-justified in a field of width 3 followed by the text you see above. The monthly pay is preceded by a dollar sign and then right justified in a field of width 10, with two decimal places.

Write a ***single*** Java statement, using **System.out.printf**, that will produce the output shown above.

**System.out.printf(**

**"%-10s %3d years old $%10.2f monthly pay\n",**

**personName, age, monthlyPay); (\n is optional)**

1. (4 pts)Write a ***single*** Java statement to declare and create an array of **Strings** named **dogNames** of length 50.

**String[] dogNames = new String[50];**

1. (3 pts)For the array **dogNames**, created in the previous step, write a ***single*** Java statement that will place the name **Eve** in the location indexed by 23.

**dogNames[23] = "Eve";**

1. (3 pts)For the array **dogNames**, created previously, write a ***single*** Java statement that returns the size of the array (i.e., the total number of elements that can be stored in the array).

**Acceptable answers include**

**dogNames.length;**

**return dogNames.length;**

**System.out.println(dogNames.length);**

1. (5 pts)Suppose we have an array list of **String** objects named **myList**. Several items have been added to the list. In the space below, write an enhanced **for** loop that will display all the entries in **myList**, with each entry on a different line.

**for(String str : myList) {**

**System.out.println(str);**

**}**