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

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

**OUTPUT**

**Leonard**

**LeonardLeonardLeonard**

**Leonard**

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

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

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

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

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

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

**public class Book {**

**private String title;**

**private boolean prizeWinner;**

**private String authorName;**

**private int pages;**

**public Book(String title) {**

**this.title = title;**

**}**

**@Override**

**public String toString() {**

**return title + " " + prizeWinner + " " +**

**authorName + " " + pages;**

**}**

**}**

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

**Book myBook = new Book("What If?");**

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

**OUTPUT**

**What If? false null 0**

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

**OUTPUT**

**7.0 7.4 7**

**int a = 37;**

**int b = 5;**

**int c = a / b;**

**double x = a / b;**

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

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

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

**OUTPUT**

**false**

**true**

**true**

**true**

**String str1 = "Leonard";**

**String str3 = "Leonard";**

**String str2 = 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 surprise. It is of type private and returns an int value. It has two parameters, the first of type double and named myDouble, and the second of type int and named myInt.

**private int surprise(double myDouble, int myInt)**

1. (4 pts)Suppose we have the following **Book** class. Read the comments preceding the no-arg constructor and add the missing code in the space provided. Do not change any existing code.

**public class Book {**

**private String title;**

**private boolean prizeWinner;**

**private String authorName;**

**private int pages;**

**public Book(String title, boolean prizeWinner,**

**String authorName, int pages) {**

**this.title = title;**

**this.prizeWinner = prizeWinner;**

**this.authorName = authorName;**

**this.pages = pages;**

**}**

**/\*\***

**\* Invokes the 4-parameter constructor. When this constructor is**

**\* used, title will be assigned the value "unknown", prizeWinner**

**\* will be assigned the value true, authorName will be assigned**

**\* the value "unknown", and pages will be assigned the value -1.**

**\* *THE BODY OF THIS CONSTRUCTOR MUST BE A SINGLE STATEMENT.***

**\*/**

**public Book() {**

**// ADD MISSING CODE HERE**

**this("unknown", true, "unknown", -1);**

**}**

**}**

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

**OUTPUT**

**6 10 6**

**6 10 7**

**6 10 8**

**6 10 9**

**WOW!**

**7 9 7**

**7 9 8**

**WOW!**

**WOW!**

**WOW!**

**int x = 5;**

**int y = 11;**

**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**

**30 10 0**

**24 15 1**

**24 20 2**

**25 0**

**19 1**

**19 2**

**20**

**Values in blue are optional**

**int a = 30;**

**for(int b = 10; 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 **name** of type **String**; the person's age in a variable named **personAge** 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 11, followed by a single space. The age is right-justified in a field of width 4 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 9, with two decimal places.

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

**System.out.printf(**

**"%-11s %4d years old $%9.2f monthly pay\n",**

**name, personAge, monthlyPay); (\n is optional)**

1. (4 pts)Write a ***single*** Java statement to declare and create an array list of **String** objects named **myList**.

**ArrayList<String> myList = new ArrayList<>(); or**

**ArrayList<String> myList = new ArrayList<String>();**

1. (3 pts)For the array list **myList**, created in the previous step, assume that 50 strings have been added. Write a ***single*** Java statement that will replace the value stored in the location indexed by 17 to **Leonard**.

**myList.set(17, "Leonard");**

1. (3 pts)For the array list **myList**, created previously, write a ***single*** Java statement that returns the number of elements currently stored in the array list.

**Acceptable answers include**

**myList.size();**

**return myList.size();**

**System.out.println(myList.size());**

1. (5 pts)Suppose we have an array of **String** objects named **dogNames** of length 50. Assume the array has been filled, so there are 50 dog names stored in the array. In the space below, write an enhanced **for** loop that will display all the entries in **dogNames**, with each entry on a different line.

**for(String dogName : dogNames) {**

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

**}**