**44-542 Object Oriented Programming Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Exam 02 (100 points) Part 2 KEY** *please print*

1. (6 pts) Complete the following statement:

In the **Object** class, for non-null reference values **x** and **y**, **x.equals(y)** is **true** if and only if \_\_\_\_\_**x == y is true**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

1. (10 pts) Suppose we have a **Dog** class that implements **Comparable<Dog>**, with the **compareTo** method comparing dogs on the basis of **registrationNumber**. In method **main** we want to sort dogs alphabetically by **dogName**. Complete the statement below so that dogs will be sorted by **dogName**.

You may assume **dogName** is of type **String** and always consists of a single word. You may also assume the **Dog** class provides a getter method for **dogName**.

Your code should fit in the space below. Do not write on the back.

**Collections.sort(myDogs, new Comparator<Dog>()**

**{**

**@Override**

**public int compare(Dog dog01, Dog dog02)**

**{**

**return dog01.getDogName().compareTo(dog02.getDogName());**

**}**

**});**

1. (6 pts) Assume method **mystery** is defined as shown here:

**public static String mystery(String str)**

**{**

**if(str.indexOf(" ") == -1)**

**{**

**return "TrickOrTreat”;**

**} else**

**{**

**return str.charAt(1) +**

**mystery(str.substring(str.indexOf(“ “) + 1));**

**}**

**}**

What is the output of the following statement?

**System.out.println(mystery("Happy Halloween To One And All"));**

**OUTPUT**

**aaonnTrickOrTreat**

1. (10 pts) Write a Java program using **try-catch** clauses to do the following: Attempt to read five integers from the file **numbers.txt**. If successful, print the string **read numbers.txt**, followed by the values of the five integers and then halt. If unsuccessful for any reason (file may not exist or the file may exist but not contain five valid integers, for example), print the string **could not obtain the data** and then halt.

Do not produce any output other that what is specified above. This means that if your program is unsuccessful for any reason, the only output should be **could not obtain the data**.

Your code should fit on the next page, which is blank except for a header line. Do not write on the back.

\*This page is blank. Write your solution to the problem on the previous page here .\*

**try**

**{**

**Scanner in = new Scanner(new File("numbers.txt"));**

**int[] numArray = new int[5];**

**for(int i = 0; i < 5; i++)**

**{**

**numArray[i] = in.nextInt();**

**}**

**System.out.println("read numbers.txt");**

**for(int i = 0; i < 5; i++)**

**{**

**System.out.print(numArray[i] + " ");**

**}**

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

**} catch (Exception ex)**

**{**

**System.out.println("could not obtain the data");**

**}**

1. (14 pts) Assume we have defined the classes and interfaces shown in the UML diagram below. Note that **Student** and **Employee** are interfaces, **AbstractStudent** is an abstract class, and all other classes are concrete.



Tell which of the following statements are legal and which are illegal. Circle the correct answer for each one.

**Student stu1 = new GraduateAssistant();** Legal Illegal

**Student stu2 = new Student();** Legal Illegal

**GraduateAssistant GA1 = new Employee();** Legal Illegal

**Employee emp1 = new GraduateAssistant();** Legal Illegal

**Employee emp2 = new Graduate();** Legal Illegal

**AbstractStudent stu3 = new Undergraduate();** Legal Illegal

**GraduateAssistant GA2 = new Graduate();** Legal Illegal

1. (4 pts) Using the same UML diagram as for the previous problem, consider the following code segment.

**Graduate grad1 = new GraduateAssistant();**

**GraduateAssistant GA3 = new GraduateAssistant();**

**GA3 = (GraduateAssistant) grad1;**

Which of the following statements is true about the code segment above? Circle the correct answer – be sure to circle only one choice; if you circle more than one choice, you answer will be considered incorrect.

The code segment will not compile.

The code segment will compile, but there will be a runtime error.

The code segment will compile and run.