Computer Science 161

Second Hour Exam

Name _____

Friday, Oct. 24, 2008
90 Pts. (will be normalized to 100 pts.)

I. Some details about objects:

Suppose that we have an Employee class with private fields empName, and salary (a double), and a constructor Employee(String eNme, double sal). The toString method has been over-ridden to make a nice String for an Employee object.

a. (5 pts.) Using the **new** command write the code necessary to specify and create a new object (call it **anEmployee**) of type Employee with name Poirot and salary 15.5.

b. (5 pts.) What two things does the **new** command do in the problem above?

c. (5 pts.) What does it mean to over-ride the toString method?

d. (5 pts.) What does the statement

System.out.println(anEmployee)

do? Include some (brief) detail

II Control structures (15 pts.)

Name the three basic control structures and give an example of each.

III. ArrayLists

a. (5 pts.) What **import** statement must be used in order to use ArrayLists?

b. (5 pts.) Write the code necessary to define **workingGroup** as an ArrayList of **Employee** (the class used above)

c. (15 pts.) Assume that the class Employee has an accessor methods getEmployeeName() and getPay() and a mutator method setPay(double newPay):

i. Write the code to get the employee at index **k** in the workingGroup ArrayList and to print that Employee's name.

ii. Write the code to delete the employee at index 3 in workingGroup.

iii. Write the code to get the employee at index **k** in workingGroup and give that employee a 10% raise.

d. (10 pts.) Use a for loop (either kind) to print out each of the employees in workingGroup (assume that some kind person has already entered them in). Remember that the toString method of Employee has been over-ridden.

e. (15 pts.) Construct an iterator for workingGroup and use it to print out each of the employees in your ArrayList. Be sure to include any **import** statements necessary.

IV. Arrays (a brief taste)

a. (5 pts.) Write the code necessary to specify **numberList** as an array of 100 integers.

b. (5 pts.) Write the code necessary to fill numberList with 100 random numbers.

c. (5 pts.) Write the code necessary to add up the elements in the array **numberList** and print the sum.