# Computer Science 161

# Third Hour Exam

## Name ________________________

Friday, Nov. 21, 2008
100 Pts

I. Some definitions and basic questions (5 pts. each)

a. What is a key?

b. What is software engineering?

c. What is unit testing?

II. Arrays.

a. (10 pts.) Suppose that the integer array x has been initialized with 100 random integers from x[0] through x[99]. Write the statements necessary to sort the array into ascending order.

b. (10 pts.) Suppose that an array of names

BROWN / CADFAEL / DREW / FANSLER / MARPLE / POIROT / WOLF

has been placed in ascending order into an array Names with BROWN at Names[0] and WOLF at Names[6]. List the names in the order in which they would be accessed in a search for DREW in

A **sequential** search

A **binary** search

## III. HashMap

a. (10 pts.) Describe some of the differences between the HashMap collection structure and the ArrayList collection structure.

Suppose now that we are writing a multi-player game, and that we want to store details of the players in a HashMap. Suppose further that we have a **Player** class, with fields playerNumber, playerName, playerEmailAddress, and playerPoints..

b. (10 pts.) Write the code to specify **players** as a HashMap of Players with **playerNumbe**r as the key.

c, (10 pts.) Write the code necessary to add a new player ("P013", "HOLMES", "holmes@ups.edu", 130) to the HashMap. Assume a constructor for Player which accepts these fields.

d. (10 pts.) Write the code necessary to print the names and email addresses of current players (that is, those currently in the **players** list).

IV. User interaction

a. (10 pts.) Write the code necessary to display a cheerful message to the user (using JOptionPane classes), and wait until the user clicks the "OK" button. A single statement suffices for this problem.

V. (15 pts.) Returning now to your established Inventory object (it does not matter whether you are using an ArrayList or a HashMap since all we are interested in for this problem is the public interface (public methods) of your Inventory class. On the next page, write a public static void main program which will do the following:

1. Create an Inventory object called myInventory.
2. Add a part to myInventory (you make up the details of the part name, etc.)
3. List the parts currently in myInventory

(space for problem V)