CS 161. Fall 2015. Midterm 1

Name:

Student ID:

1:

2:

3:

4:

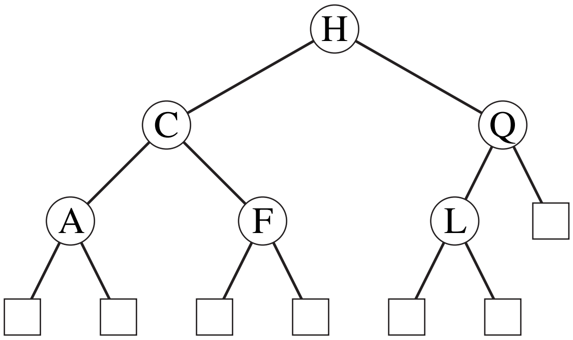
5:

Total:

1. (25 points) Suppose that we merge the two lists X and Y given below, each of which is already sorted in numerical order. List the pairs of numbers that will be compared with each other, in the order in which they are compared.

$$X = [10, 28, 65, 89] \qquad Y = [20, 25, 41, 47]$$

2. (25 points) The drawing below shows a WAVL tree, with circles for internal nodes (labeled with each node's key) and with squares for external nodes. The ranks of the nodes are not shown. What value or values are possible for the rank of the root node?



3. (20 points) Is radix sorting stable? Explain why or why not.

4. (30 points) You have been hired to help program a piece of software that counts votes for a new award for the best new Android app developed by an ICS student. As part of this software, you need to write a subroutine takes as input a list L of strings (names of apps from the votes) and produces as output a dictionary of key-value pairs where the keys are the names of the apps and the values are the numbers of votes for each app. Another programmer has been asked to implement a linear-probing hash table for you to use. It has four operations: initialize a new dictionary, find whether a key is in a dictionary, look up the value associated with a given key, and add a given key-value pair to the dictionary (replacing any existing pair that has the same key). For example, for the input

   ["Happy", "Wrapper", "Happy", "Approach", "Wrapper", "Happy"]

   you should return as your output a dictionary containing the three key-value pairs

   ("Approach", 1), ("Happy", 3), and ("Wrapper", 2).

   Give pseudocode for your subroutine. For full credit, your pseudocode should be no more than ten lines long and should run in linear expected time. Do not describe how to implement a dictionary, and do not use any other dictionary operations than the four listed above. Do not include any variable declarations; your answer should be pseudocode, not code.

5. (20 points) For each number given below, how many comparisons would be needed in the worst case for any comparison tree that has that many distinct outputs, regardless of the problem being solved by the comparison tree? Your answer to each part should be an integer number of comparisons.

   (a) 64 outputs

   (b) 4 outputs

   (c) 99 outputs

You may use this page (or the back of the other pages) as scratch paper.