

**Written Questions (80 marks) + 10 marks bonus**

**Q1.** Draw the 11-entry hash table that results from using the hash function  $h(k) = (3k - 5) \bmod 11$  to hash the keys 31, 45, 14, 89, 24, 95, 12, 38, 27, 16, and 25, assuming that collisions are handled in the following ways:

- a) linear probing
- b) double hashing using the secondary hash function  $h'(k) = 5 - (k \bmod 5)$ .

**Q2.** Answer the following questions:

- a) Show (diagrammatically in the form of a divide-and-conquer tree, in the same format shown in your text book and slides) all the steps involved in the **quick sort** on the following input sequence: 22 72 38 48 13 14 93 69 45 58 13 81 79. You should sort the sequence in the non-decreasing order. Clearly show the pivots chosen at each node. (Note: Though in practice the pivot for each input is chosen at random, you may choose pivots here of your choice which make the tree balanced, i.e. only choose *good* pivots).
- b) Repeat part a) if the quick sort is performed in place.

**Q3.** Answer the following questions:

- a) Explain why Merge sort is the most suited for very large inputs (that do not fit inside memory) while quick sort and heap sort are not as suited. Note that these three sorting techniques have comparable time complexities.
- b) Can Merge sort be performed in place? Explain your understanding.

**Q4.** Show the steps in performing radix sort on the following input: 69 81 45 48 13 38 58 9 14 22 93 79 5 72. You should sort the sequence in the non-decreasing order.

**Q5.** Answer the following questions:

- a) Consider the following sequence of keys inserted into an initially empty AVL tree T in this specific order: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11. Draw the AVL tree T after each insertion.
- b) Is an AVL tree for a specific set of inputs unique? If your answer is yes, then explain why it is so. If your answer is no then demonstrate using the set of inputs in part a) above.

**Q6.** Answer the following questions:

- a) A team of biologists keeps information about DNA structures in an AVL tree using as key the specific weight (an integer) of a structure. The biologists routinely ask questions of the type: "Are there any structures in the tree with specific weights between  $a$  and  $b$  (both inclusive)", and they hope to get an answer as soon as possible. Design an efficient algorithm that given integers  $a$  and  $b$  returns true if there is a key  $x$  in the tree such that  $a \leq x \leq b$ , and returns false if no such key exists. Describe your algorithm in pseudo-code.
- b) What (and why) is the time complexity of the algorithm?

**Q7.** Describe an efficient algorithm for computing the height of a given AVL tree. Your algorithm should run in time  $O(\log n)$  on an AVL tree of size  $n$ . In the pseudocode, use the following

terminology:  $T.left$ ,  $T.right$ , and  $T.parent$  indicate the left child, right child, and parent of a node  $T$  and  $T.balance$  indicates its balance factor (-1, 0, or 1).

For example if  $T$  is the root we have  $T.parent = \text{nil}$  and if  $T$  is a leaf we have  $T.left$  and  $T.right$  equal to  $\text{nil}$ . The input is the root of the AVL tree. Justify correctness of the algorithm and provide a brief justification of the runtime.

**Q8.** Given a balanced binary search tree that somehow allows duplicates populated by a sequence  $S$  of  $n$  elements on which a total order relation is defined, describe an efficient algorithm for determining whether there are two equal elements in  $S$ . What is the running time of your algorithm?

**Q9\*) BONUS** Given the following hash functions, what is the probability that two randomly chosen keys collide? (justify your answer).

- a)  $h(k) = k \bmod 13$
- b)  $h(k) = 2k \bmod 8$
- c)  $h(k) = (3k+1) \bmod 12$
- d)  $h(k) = (((5k+7) \bmod 13)+1) \bmod 17$
- e)  $h(k) = (k \bmod 6)^2$