Indian Statistical Institute
CS3
Third Year Students
14Apr2022
Closed Book Exam – Duration 3 hours.

Total Marks: 60

ANSWER Q1, any three parts of Q2 and EITHER Q3 OR Q4.

Q1. [Total Marks:1.5x(first)6 +2.5x(rest)6 = 24]

Write short answers to the following questions. No derivation/explanation is necessary

a.) True or False? If f(n) is $O(n^3)$ and g(n) is $O(n^3)$ then f(n)/g(n) is O(1)

b.) True or False? If f(n) +g(n) is $O(n^2)$, then f(n) is $O(n^2)$

c.) True or False? Given two BSTs, to determine if they are identical, it is enough to compare their inorder traversals.

d.) True or False? Given a Binary tree with integer data, an inorder traversal will output the data in sorted order.

e.) True or False? An array of integers is turned into a Max_Heap using Max_Heapify and Build_Max_Heap procedure. The resulting tree is height balanced.

f.) An AVL tree has initially n elements. Additional n^2 elements have to be inserted. What is the best bound on the time complexity? $O(n^2)$, O(n lg n), $O(n^4)$, $O(n^2 \lg n)$, $O(n^3)$ (Select one option)

g.) Fill in the gap appropriately: Merge sort guarantees to sort an array of N items in time O(N lg N), its prime disadvantage is that it uses --- --- of the order ….. (Options for the second gap are O(lg N), O(N), O(N lgN))

h.) Choose the correct option: In recursive merge sort, data comparisons are done (after/before) recursive calls are complete.

i.) Fill in the gaps: The minimum and maximum number of nodes for an AVL tree of height 5 are …. and …. respectively.

j.) A BST and a Max_Heap is built from the same array of integers. Which one is likely to have greater height?

k.) An almost complete binary tree is made out of the array 30, 10, 50, 20, 90, 60, 70, 80 and then it is turned into a Max Heap following Max_Heapify and Build_Max_Heap. At this point, what is the arrangement of data in the array?

l.) Complete the lps array [0, 1, 0, 1, 2, - , -, -, -, -,] for the pattern "AABAACAABAA" (lps is the largest prefix suffix array in the context of KMP algorithm)

Q2. [Total Marks: 3x5=15]

DO ANY THREE from 2a, 2b,2c, and 2d

a.) Prove that in any algorithm that uses comparisons as a way to search for an element within a set of n elements, ordered or unordered, the best time complexity that can be achieved is lg n.

b.) Determine with justification the order of complexity (O(…)) of creating a BST from any sequence of n integers while maintaining AVL property. You can assume that maximum height of an AVL tree is O(lg n).

c.) In KMP pattern matching algorithm, the function pi[p] or lps[p] is a precalculated array whose elements are the lengths of the largest prefix that is also a suffix for each substring of the pattern "p". For example, if "p" is "aba" then lps[p] is {0,0,1}. Note that lps[p] starts with 0.

Explain why the {0,1,2,4,0} cannot be the output of lps[p] for any pattern string "p". State and prove the corresponding theorem.

d.) Given an array A of integers, we know how to create an almost complete binary tree. Let Max_Heapify(A,i) be the function that transforms the subtree rooted at the index i into a max_heap provided that that subtrees with roots at left[i] and right[i] are max heaps.
Write an algorithm for Max_Heapify(A,i). What is the complexity of Max_Heapify?

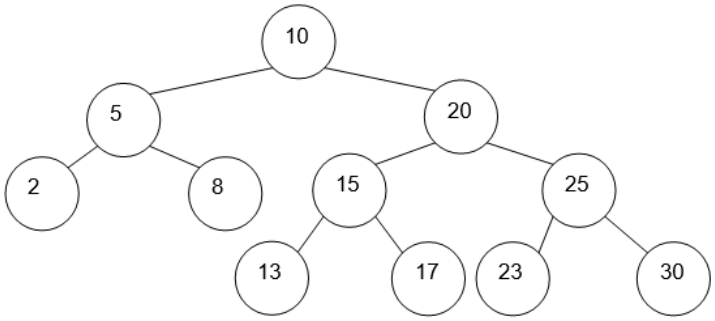
ANSWER EITHER Q3 OR Q4 (NOT BOTH).

Q3. [Total Marks:3+3+7+8 =21]

a.) Draw an AVL tree of height 4 that has the minimum possible number of nodes. Is your answer unique?

b.) Show the BST tree created out of 1,2,3,4,5,6,7 while maintaining AVL property. (Show ONLY the final AVL BST, not intermediate steps.)

c.) For the following AVL tree:



I. Label each node above with its Balance Factor.
II. What range of numbers can be inserted to cause a right-right imbalance, and at which node does the imbalance occur?
III. What value can be inserted to cause a right-left imbalance? At which node does the imbalance occur?

d.) Insert 18 into the above given AVL tree. Relabel the balance factors. What type of imbalance does it cause? Show the result after balancing.(show intermediate diagrams)

Q4.[Total Marks: 5+3+8+5=21]

a.) Prove that if you have a large enough data set, there will always be a subset of data that will map into the same location in a hash table no matter how good the hash function is.
b.) Explain the concept of Probing function in the context of collision resolution of a hash function.

c.) Generate the TWO hash tables of size 10 (for linear and quadratic probing function) for the set of integers {89, 18, 49, 58, 69} when the hash function is given by h(x) = x mod 10 and collision resolved by

$h_i(x) = (h(x)+i) \bmod 10$ (linear probing function)

$h_i(x) = (h(x)+i^2) \bmod 10$ (quadratic probing function).

YOU NEED TO JUST PRODUCE THE TWO Hash TABLES, INTERMEDIATE CALCULATIONS ARE NOT REQUIRED.

Using these tables, compare the pros and cons of a linear and a quadratic probe function.

d.) If the table size is prime and the table is at least half empty, then prove that quadratic probing will always find an empty location.