Prof. Pietro S. Oliveto

CS217 Data Structures and Algorithm Analysis
Fall term 2023

Exercise Sheet 9

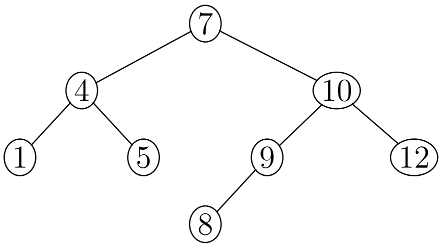
Handout: Nov 14th — Deadline: Nov 28th, 4pm

Question 9.1 (1 mark)

1. Prove by induction that every complete binary tree of height h has $2^h - 1$ internal nodes.
2. Prove by induction that in every full nonempty binary tree the number of leaves is one more than the number of internal nodes.
3. Prove by induction that every nonempty binary tree satisfies $|V| = |E| + 1$.

Question 9.2 (0.25 marks)

1. Insert a node with key 11 into the following binary search tree. Give a step-by-step explanation.



2. Delete the node with key 10 into the resulting binary search tree. Give a step-by-step explanation.
3. Insert a node with key 10 into the resulting binary search tree. Give a step-by-step explanation.
4. Delete the node with key 8 from the resulting binary search tree. Give a step-by-step explanation.
5. Delete the node with key 7 from the resulting binary search tree. Give a step-by-step explanation.

Question 9.3 (0.25 marks)

Delete two different nodes in different order from a binary search tree (e.g. first node x and then node y, or alternatively first node y and then node x). Can the resulting trees be different? Explain your answer.

Question 9.4 (0.25 marks)

Write the Tree-Predecessor(x) procedure.

Question 9.5 (0.25 marks)

You can sort a set of n numbers by the following procedure:

1. Build a binary search tree by inserting each element using TREE-INSERT (n times)
2. Print the numbers in sorted order by an INORDER tree walk.

What are the worst case and best case runtimes of this sorting algorithm?

Question 9.6 (1 mark)

1. Implement a Binary Tree using a linked list (as explained during the lecture) to encode a mathematical expression with binary operators (+,-,*,/) provided in input in prefix notation i.e., functional programming notation. You should use a stack to keep track of the pointers of the nodes that you will need later to fill in their right child.
2. Implement the procedures INORDER, PREORDER and POSTORDER and print the respective outputs when applied to the resulting tree from Step 1.

Example input: $- + a * bc/de$ which leads to the tree given in the lecture slides.