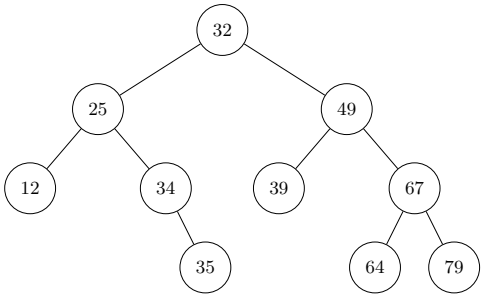
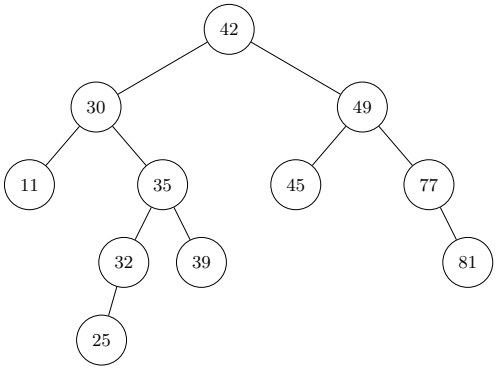
XJTLU | Computing

Trees

Questions

Problem 1. State whether the following trees are binary search trees (BSTs) or not and why.

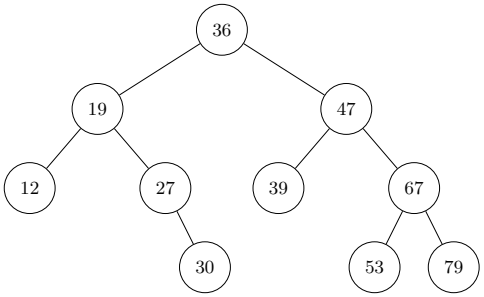




Problem 2. Starting from an empty binary search tree (BST), insert using the classic algorithm, one after the other, the sequence of keys 32, 15, 47, 67, 78, 39, 39, 63, 21, 12, 27.

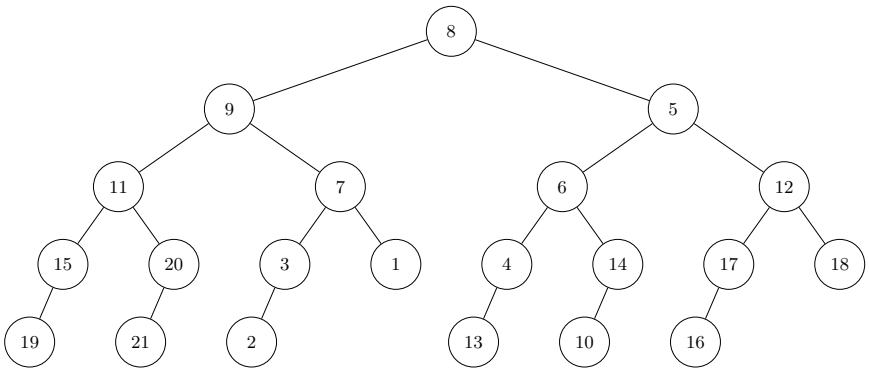
Problem 3. Explain if it is true or false that in a non-empty binary search tree (BST), the maximum element can have a left child node, but not a right one.

Problem 4. Starting from the following binary search tree (BST), eliminate the keys 53, 27, 19, 36 one after the other. Explain which algorithm you used to eliminate the keys.



Problem 5. Draw the binary search tree (BST) that results in the following preorder traversal: 8, 5, 2, 1, 4, 3, 6, 7, 11, 9, 13, 12.

Problem 6. What are the sequences of nodes encountered when traversing the tree below in preorder, postorder, and inorder?



Problem 7. Consider the following two orders for traversing binary search tree (BST).

a1. Traverse the right subtree
a2. Visit the root
a3. Traverse the left subtree

b1. Visit the root
b2. Traverse the right subtree
b3. Traverse the left subtree

Are there any simple relationships between the sequences of nodes encountered following these orders and those generated by the three orders that we mentioned in the lecture?

References

Atserias, Albert et al. (2022). *Data Structure and Algorithms Problem Set*. Universitat Politécnica de Catalunya.

Wirth, Niklaus (1978). *Algorithms + Data Structures = Programs*. (Oberon version: August 2004). Upper Saddle River, NJ, USA: Prentice Hall. ISBN: 0130224189.