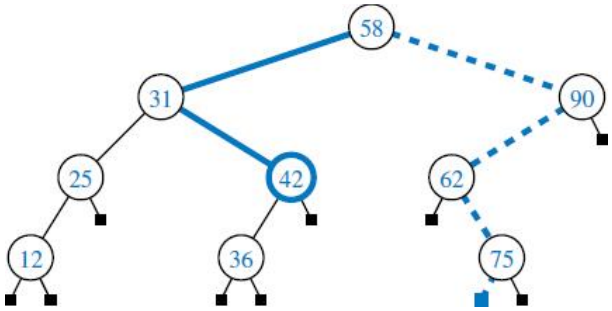
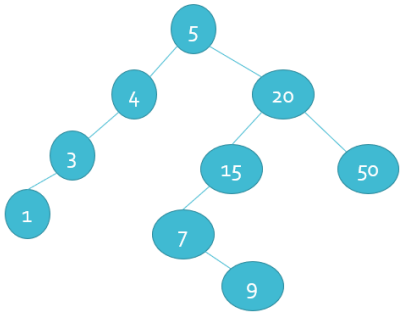
Lab 9



Example 1

- Implement the BinaryTree data structure
- In addition to the update method explained in the lectures, implement a method named "add" that allow us to build a binary search tree.
- **The binary search tree** is a binary tree in which the elements are stored so that the given subtree the elements of its left subtree are smaller than the element stored in the root which itself smaller than the elements in the right subtree. See the tree in "Example 1" image above.
- The **'add'** method should be implemented as follows:
    - It receives an element (for simplicity we can assume that we're only working with integers)
    - If the tree is empty the given element should be added as the root.
    - If the tree is not empty then starting at the root
        - If the value of the given element is less than the element of the node, add the new element to the left-subtree.
        - Otherwise, add the new element to the right-subtree.
- For example, if we add the values {5, 4, 20, 15, 3, 1, 50, 7, 9} in the given order. The tree should look like 'Example 2' image below



Example 2

- When you print the list based on the inorder traversal it should print the values in proper order: {1, 3, 4, 5, 7, 9, 15, 20, 50}
- Implement another method named **'find'** that takes a value and returns true if there's a node in the tree that contain a similar value and false otherwise.

    - If the tree is empty it returns false
    - Starting from the root,
        - if the given value is equal to the element in the node it return true
        - otherwise,
            - if the given value is less than the element of the node, it searches for the value in the left subtree
            - if the given value is greater than the element of the node, it search for the value in the right subtree
    - When testing the find method on the tree in 'Example 2', calling find(7) should return true but calling find(100) should return false.
- Include comments with both methods to explain the code **in your own words.**

Submission Instructions

- Submit your solution in a word or pdf document
- Late submissions will be accepted for up to 3 days late with 10% late penalty for each day.
- You are expected to work on this assignment as an individual. Any violation to academic integrity will result in a zero mark.
    - I understand that some students find a sort of unhealthy fulfillment in cheating the system, but you'd find much better fulfillment if instead you put your energy into building real competence that will help you get a good job with high payment.

How is the lab conducted?

- The lab assignment will be posted on Friday and will be due on the next Friday (24hrs after the last lab)
- If you have any question
    - Imagine you'd win 1000$ if you figured it out on your own and try it first (you'd actually win much more since it's a skill that you'd need throughout your career)
    - If you couldn't figure it out and you're ready to lose the 1000$
        - Join the Zoom meeting of your lab (only the one you registered for)
        - Raise your hand and wait for the TA to take your question
- It's better for you to work on it and submit it during your allocated lab time (or 24hrs after), because you have other things to do, and last-minute questions are not likely to be addressed