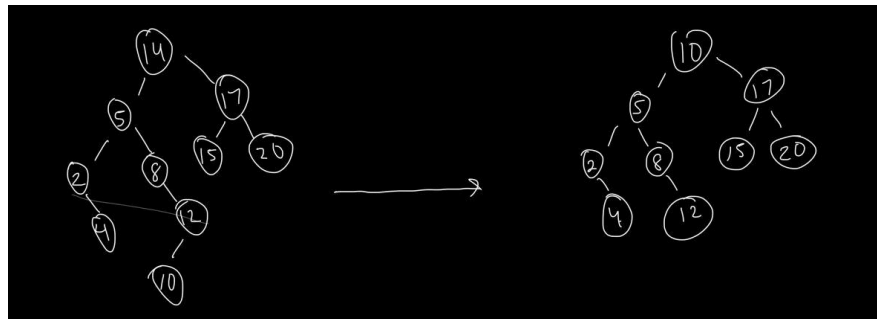


1) Recursive Programming

- a) Base case is the stopping condition, this case stops the recursion, and uses conditional if statements. The recursive case is the general scenario where the problem is broken down. Conditional if statements are used to define when to call the function again with a smaller version of the problem. The recursive case brings the problem down closer and closer to the base case.
- b) Multiple recursion is where the function calls itself multiple times within its definition. This can cause inefficiencies within the program. In order to combat this you store solved subproblems in a table and the function is able to retrieve existing solutions instead of having to recalculate them, causing a boost in performance.

2) Binary Tree

- a) The leaf nodes are 4,10,15,20
- b) Traversals
 - i) 2,4,5,10,12,8,14,15,17,20
 - ii) 4,2,10,12,5,15,20,17,14
- c)
 - i) If the node with value 14 is deleted then you would have to find a logical predecessor. You find the logical predecessor by looking at the left side of the tree and then finding the most right node value. This node value will take the place of the old root node.



ii)