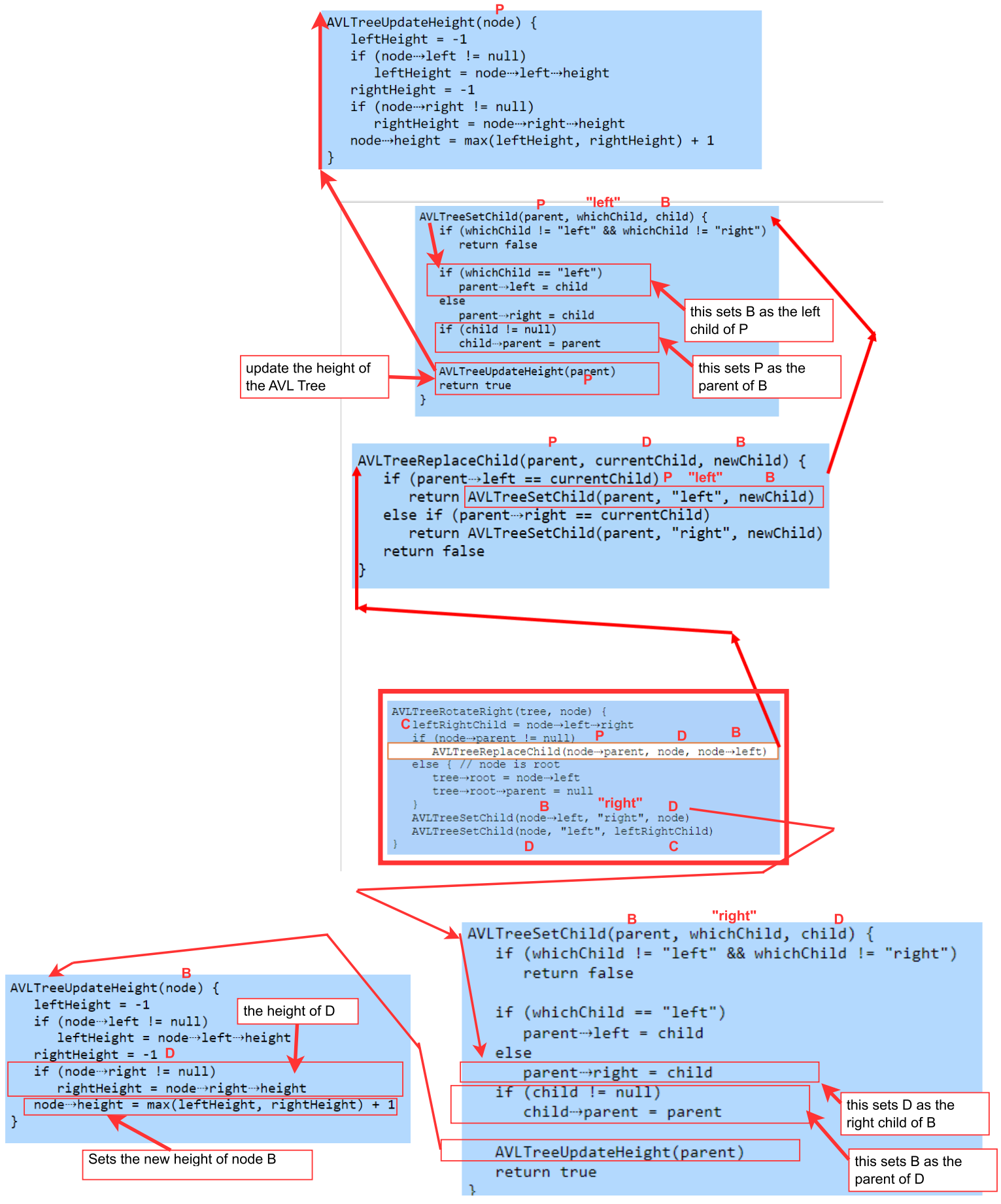
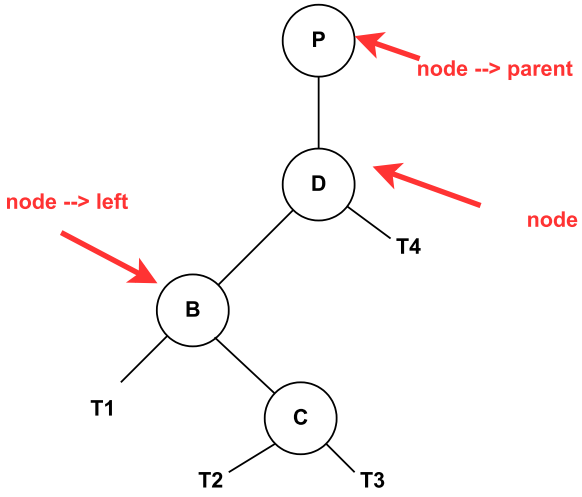
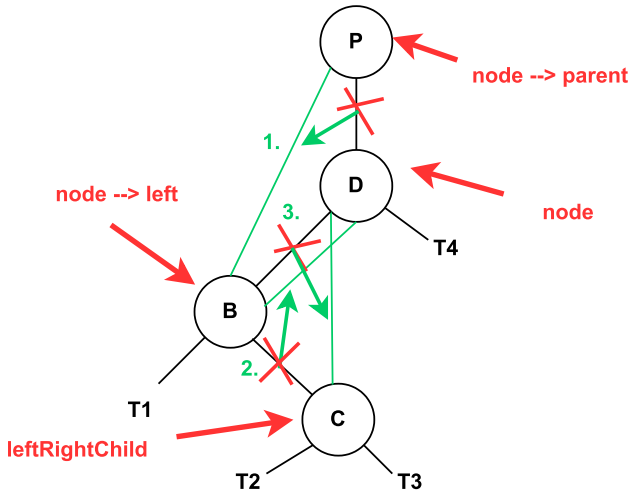
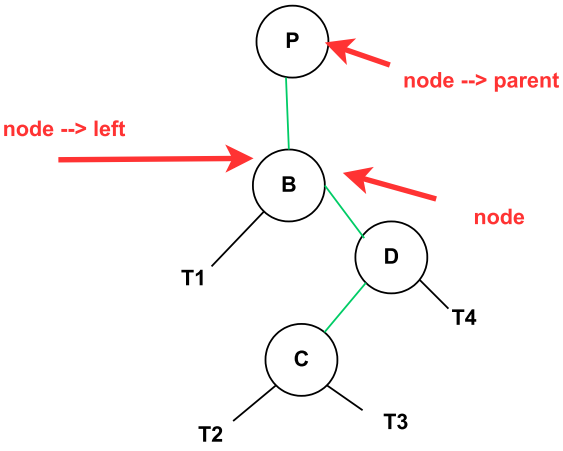
AVLTreeRotateRight(tree,node D)



AVLTreeRotateRight(tree,node D)







Insertion with Rebalancing

- An AVL tree insertion involves
  - Searching for the insert location
  - Inserting the new node
  - Updating balance factors
  - Rebalancing
- Balance factor updates are only needed on the nodes ascending along the path from the inserted node up to the root

MyNotes:
- For n nodes, an AVL tree has a height equal to the floor(log(n))
- For n nodes, an AVL tree has height O(log(n)
- For AVL insert operations the complexity is O(log(n))

Red-black tree: A balanced tree

- Every node is colored either red or black
- The root node is black
- A red node's children cannot be red
- A null child is considered to be a black leaf node
- All paths from a node to any null leaf desecendant node must have the same number of black nodes