

GIT Department Of Computer Engineering
CSE 222/505 - Spring 2020
Homework 7 Part 1

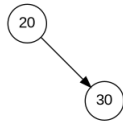
Fatih Kaan Salgır
171044009

1 AVL Tree

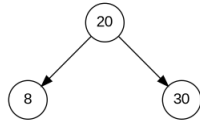
1.1 Adding



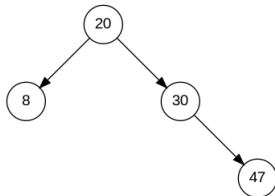
Adding first node



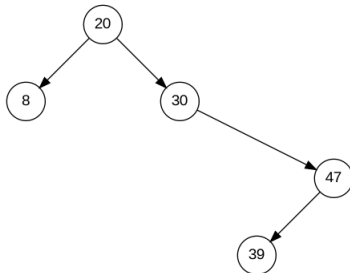
30 is greater than 20, add to right



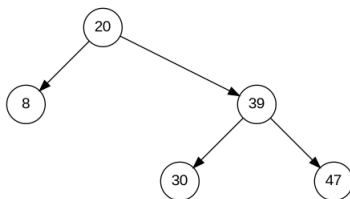
8 is less than 20, add to left



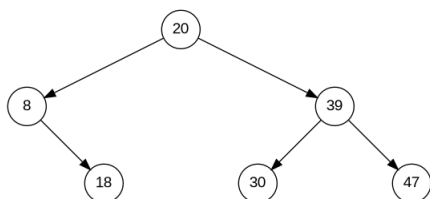
47, is greater than 20 and 30, add to right of 30



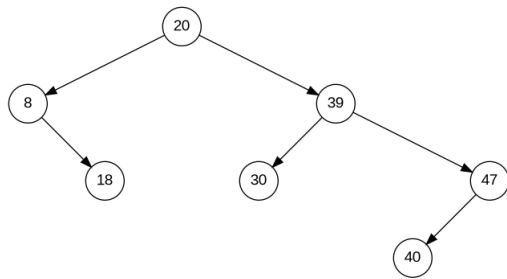
39, greater than 20 added to left of 47
Rotation needed



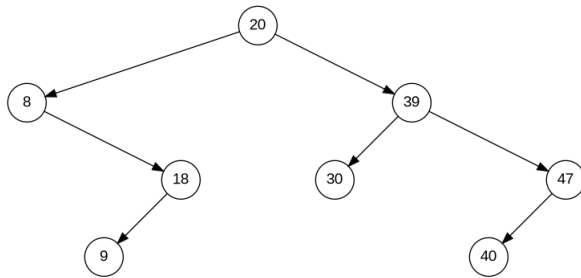
Right-left rotation performed on right sub-tree



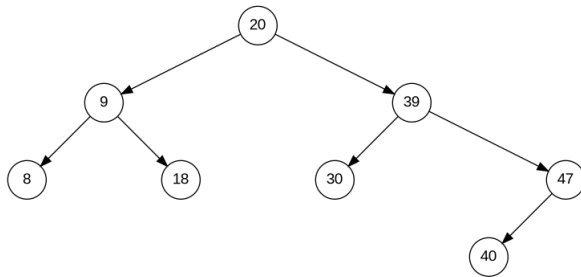
18 is less than 20, greater than 8, added right of 8



40 is greater than 20 and 39, less than 47,
added left of 40

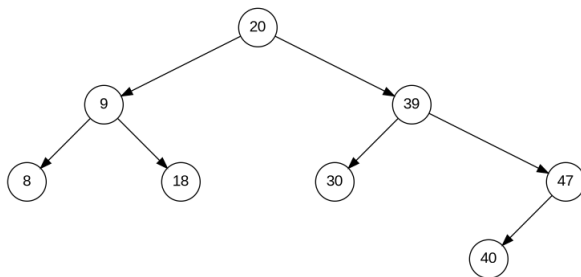


9 is less than 20, greater than 8, less than 18,
added left of 18
Rotation needed



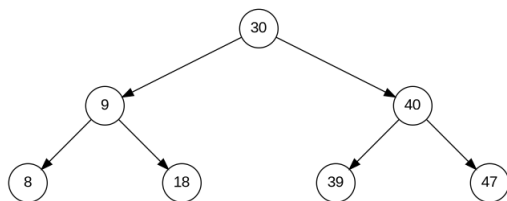
Right-left rotation performed on left sub-tree

1.2 Removing



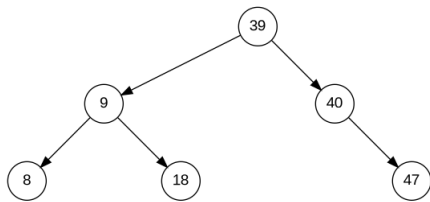
20 is to be removed, has two children, inorder
successor is 30

Left-right rotation performed to balance left
sub-tree.

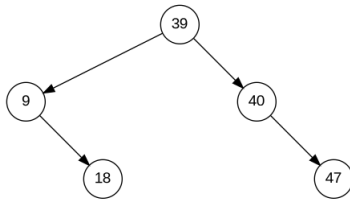


30 is to be removed, has two children, inorder
successor is 39

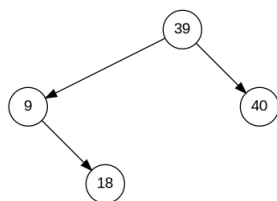
Left-left rotation performed to balance left
sub-tree.



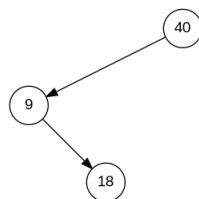
8 is to be removed, has no children, rotation not needed.



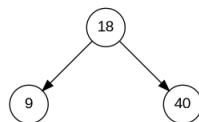
47 is to be removed, has no children, rotation not needed.



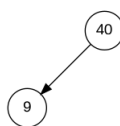
39 is to be removed, has two children, inorder successor is 40, rotation not needed.



18 is to be removed, has two children, inorder successor is 40, rotation needed.



Left-right rotation performed.



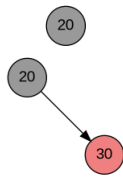
40 is to be removed, has one children, left node will be the new root.



9 is to be removed, has no children.

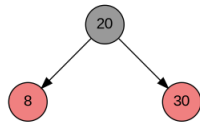
2 Red-Black Tree

2.1 Adding

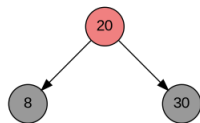


Adding first node as black.

30 is greater than 20 added to right, parent is black, done.



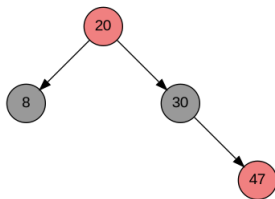
8 is less than 20, added to left, parent is black, done.



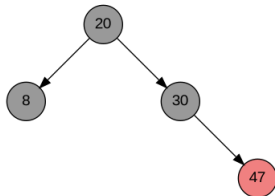
47 is to be added, since red node cannot have a red child:

Color of parent (30) and parent's sibling (8) changed to black.

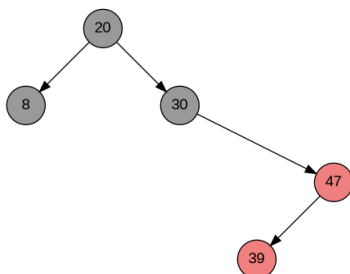
Grandparent's (20) color changed to red.



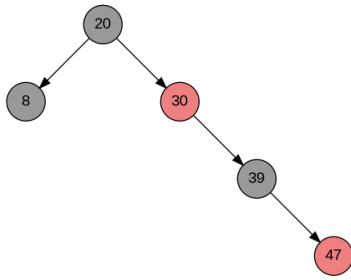
47 can be added now, but root must be black.



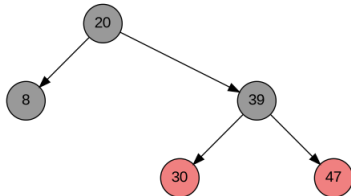
Color of root node changed to black.



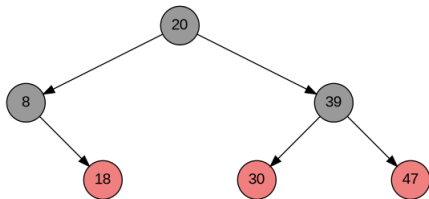
39 is to be added, BST insertion performed, but red node cannot have red child. Double rotation needed



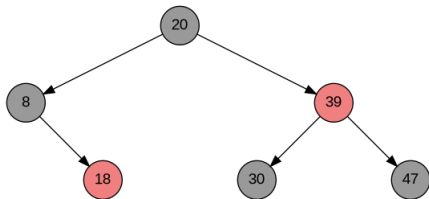
First rotation is performed between 47 and 39, color of 39 changed to black.



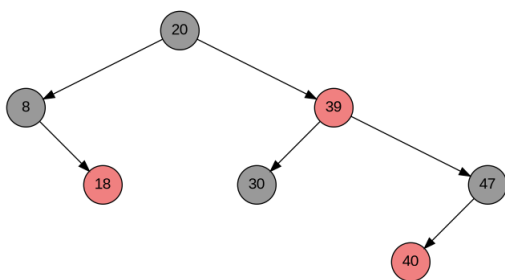
Second rotation is performed on right sub-tree.



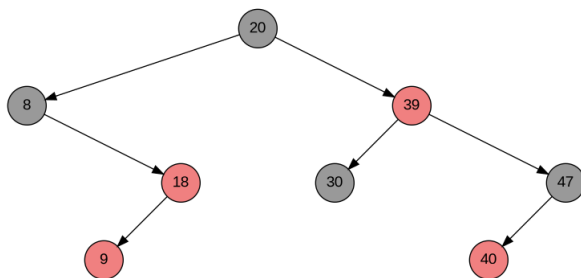
18 is to be added, BST insertion, no violation



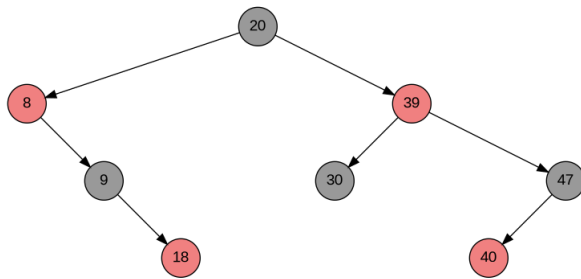
40 is to be added, but red node cannot have a red child:
Color of parent (47) and parent's sibling (30) changed to black.



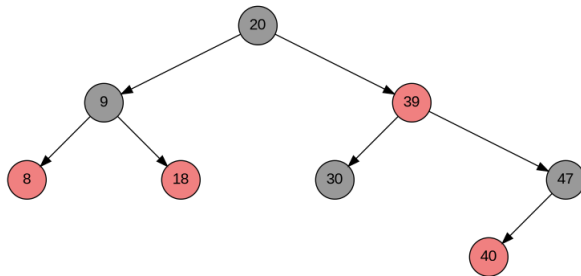
40 can be added now, BST insertion performed.



9 is to be added, BST insertion performed, it violates the rule: red node can only have black child. Double rotation needed.

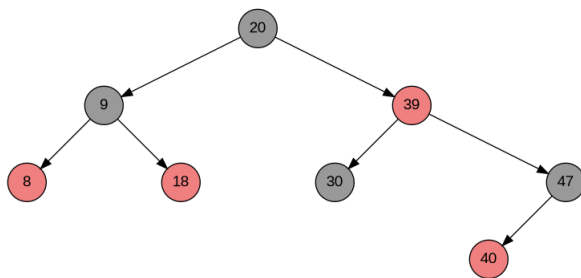


First rotation performed between 9 and 18. Color of 9 changed to black.

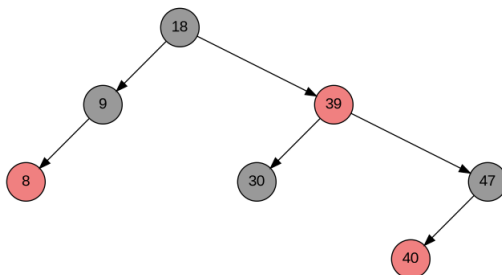


Second rotation performed on left sub-tree.

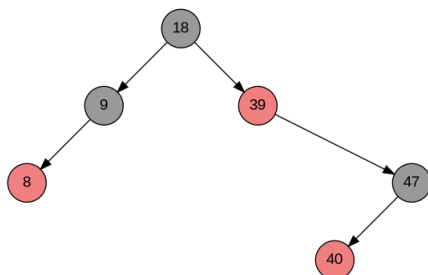
2.2 Removing



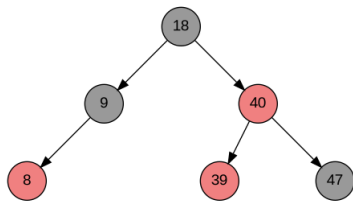
20 is to be removed, inorder predecessor is 18, 18 will take place of 20, color of 18 changed to black to not to violate the rule: Root node must be black.



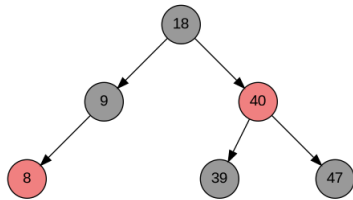
30 is to be removed, BST removal performed.



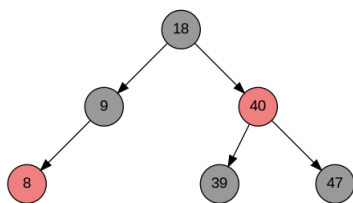
Rotation needed on right sub-tree. 40 will be parent of 39 and 47.



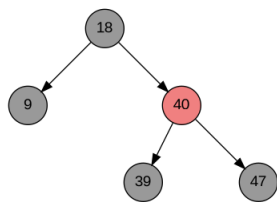
Rotation preformed, but red node cannot have a red child.



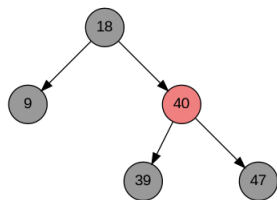
Color of 39 changed to black.



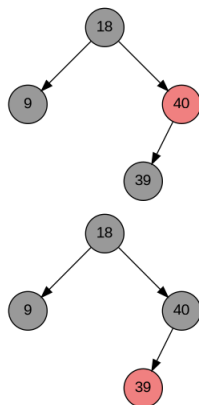
8 is to be removed.



BST removal of 8, it is red node, done.

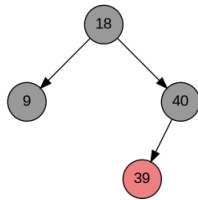


47 is to be removed.

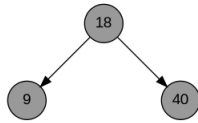


BST removal of 47. Colors need to be swapped.

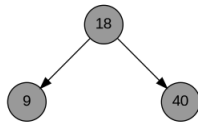
Color of 40 set to black, color of 39 set to red.



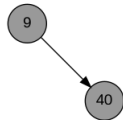
39 is to be removed.



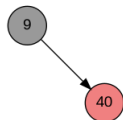
BST removal of 39, it is red node, done.



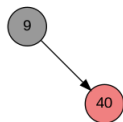
18 is to be removed. Predecessor is 9.



9 took place of 18. Since there is only one child, it must be red.



Color of 40 changed to red.



40 is to be removed.



BST removal of 40, it is red node, done.



9 is to be removed. Deletion of root node will be performed.

3 2-3 Tree

3.1 Adding



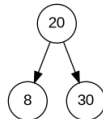
Addition of first node.



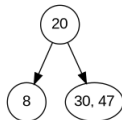
30 is to be added, 30 is greater than 20, added to same node.



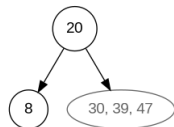
8 is to be added, virtually added to node. Need rearrangement.



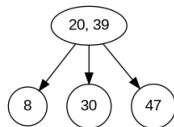
Root node seperated.



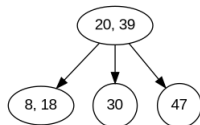
47 is to be added, greater than 20, greater than 30, added the same node as 30.



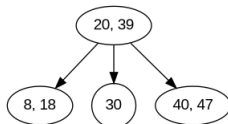
39 is to be added, greater than 20, virtually added between 30 and 47. Need rearrangement.



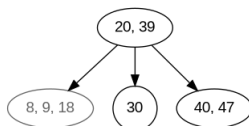
39 moved to parent node.



18 is to be added, less than 20, added the same node as 18.



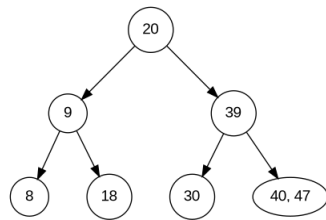
40 is to be added, greater than 20 and 39. Added as the same node as 47.



9 is to be added, less than 20 and 39. Virtually added as the same node as 8 and 18.

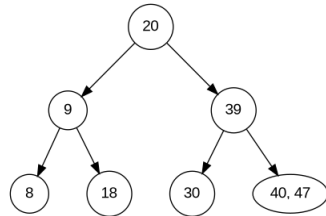


9 is moved to parent node, but parent node has 3 elements now. Need rearrangement.

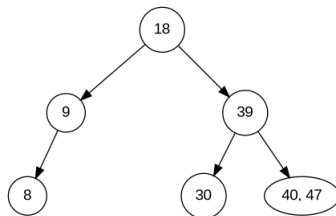


Root node separated

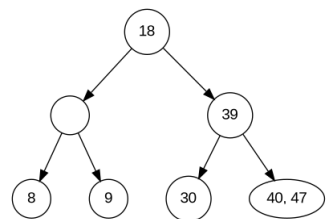
3.2 Removing



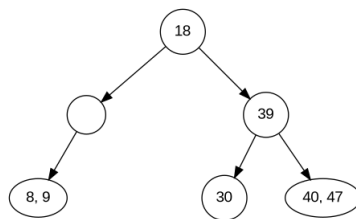
20 is to be removed, predecessor is 18.



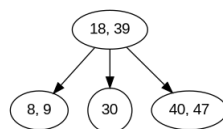
18 took place of 20.



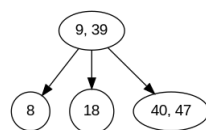
9 moved to child node.



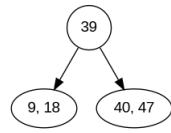
8 and 9 merged.



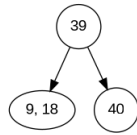
18 moved to child node, and merged with 39.



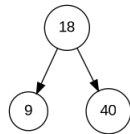
30 is removed, 18 moved to middle child, 9 moved to parent node.



8 is removed, 9 is moved to left child, left child and middle child merged.



47 is removed.



39 is removed. 18 moved to parent node.



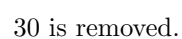
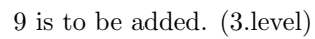
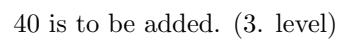
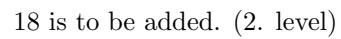
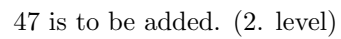
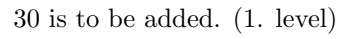
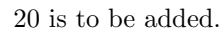
18 is removed, 9 and 40 merged.

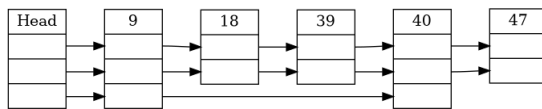


40 is removed.

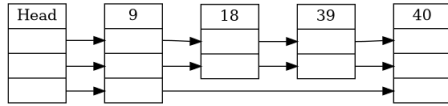
4 Skip List

4.1 Adding

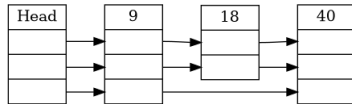




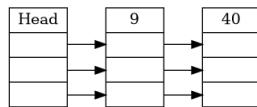
8 is removed.



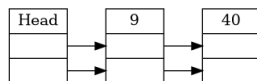
47 is removed.



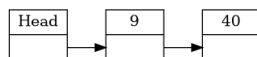
39 is removed



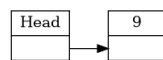
18 is removed.



Level 3 is not needed, removed.



Level 2 is not needed, removed.



40 is removed



9 is removed.

5 B-Tree with order 4

5.1 Adding



20 is added.



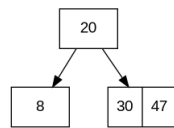
30 is greater than 20, added right of 20.



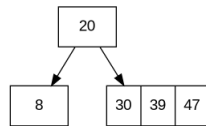
8, is less than 20, added left of 20.



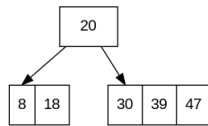
47, is added the root node virtually.



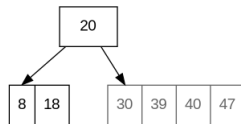
Nodes are seperated.



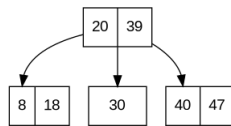
39, is greater than 20, added to right child of 20.



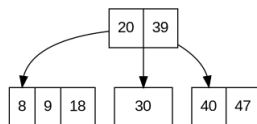
18, is less than 20, added to left child of 20.



40, is greater than 20, added to left child of 20.

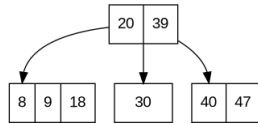


Nodes are seperated.

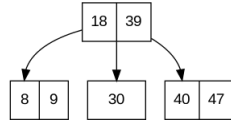


9, is less than 20, added to left child.

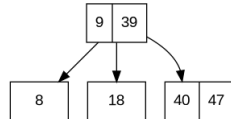
5.2 Removing



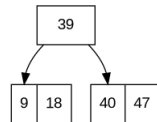
20 is to be removed, inorder predecessor is 18.



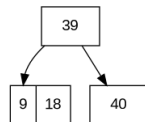
18 took place of 20.



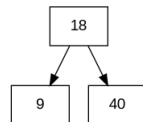
30 is removed, 18 moved to middle child. 9 moved to parent.



8 is removed, 18 is merged with left sibling. 9 is moved to left child.



47 is removed.



39 is removed, 18 moved to parent.



18 is removed, 9 and 40 merged together.



40 is removed.