Deakin University

Data Structures and Algorithms

OnTrack Submission

AVL-Trees

Submitted By:
Iftekhar Qureshi
qureshii
2020/09/30 16:38

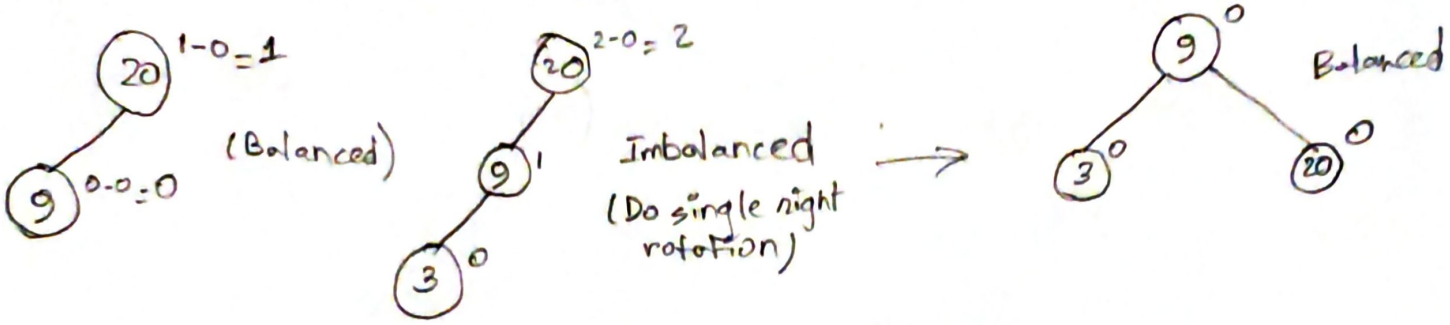
Tutor:
Maksym Slavnenko

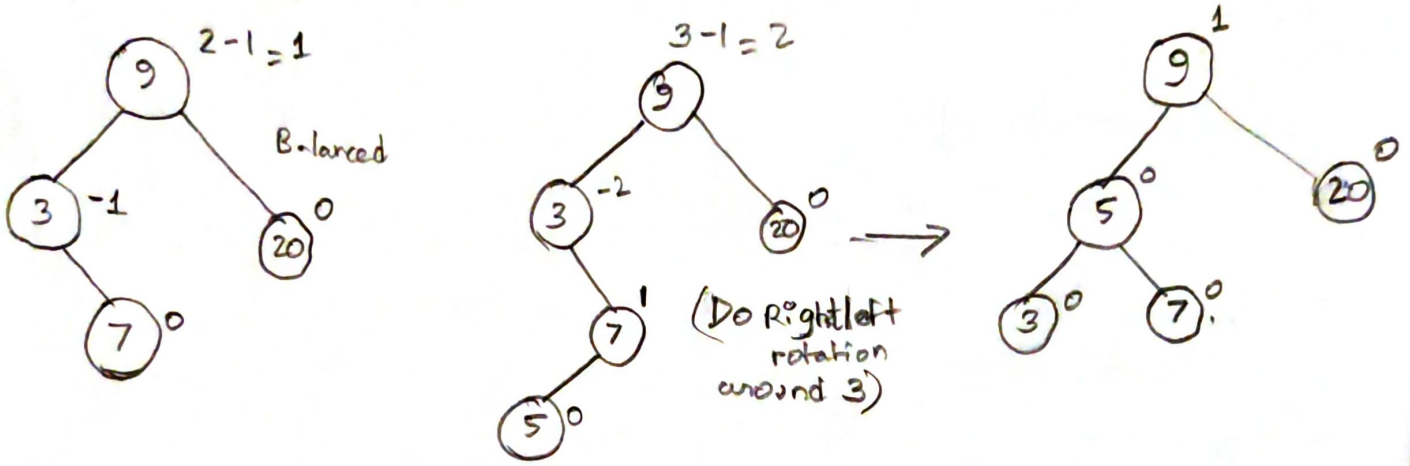
Outcome	**Weight**
Implement Solutions	♦◊◊◊◊

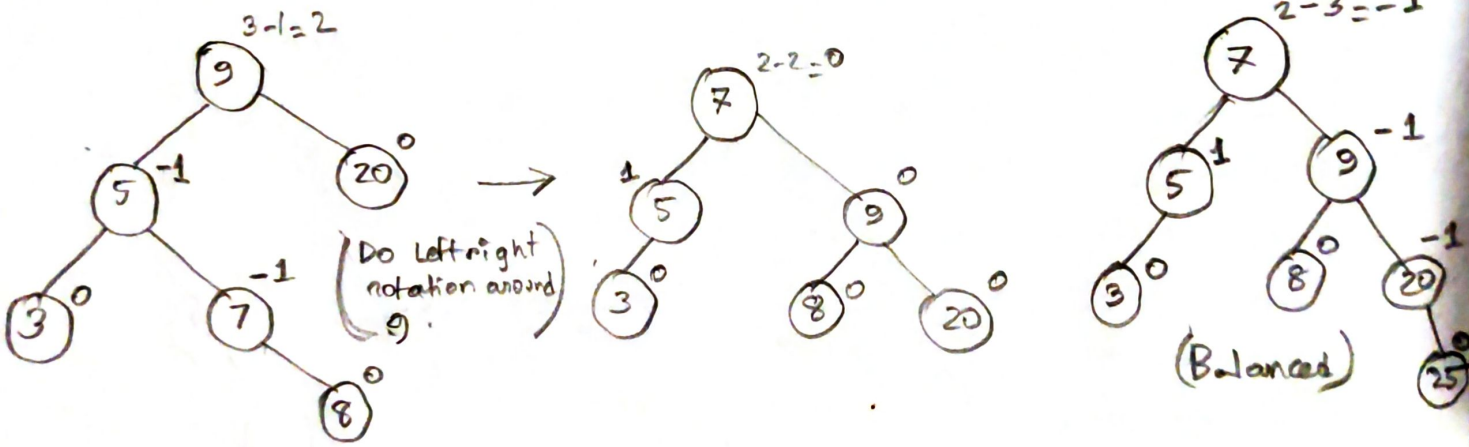
This task was to help us learn the concept behind AVL trees which is a type of binary search tree.

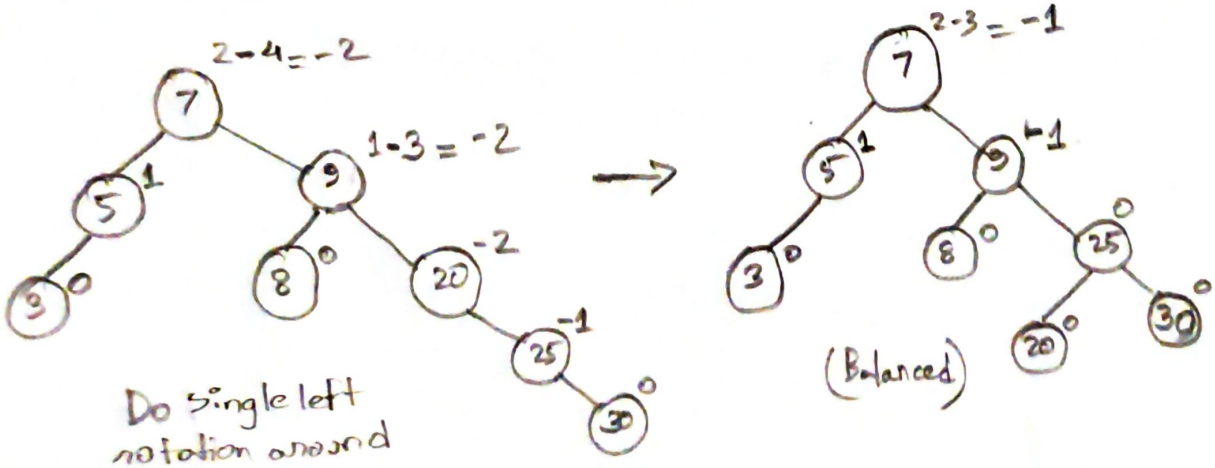
September 30, 2020

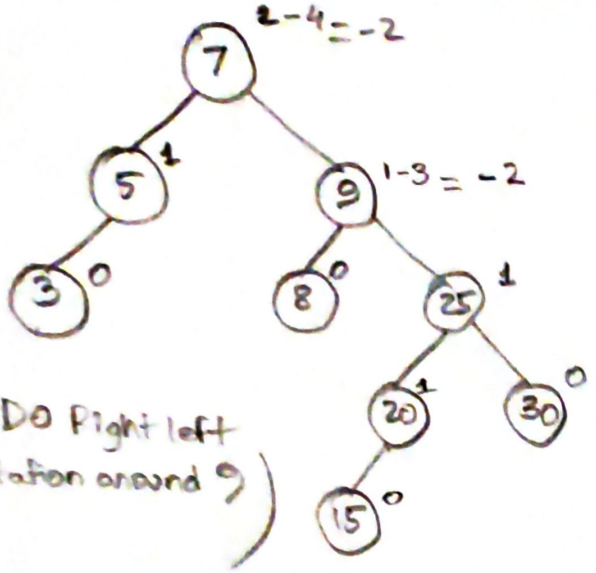
Insertion- 20, 9, 3, 7, 5, 8, 25, 30, 15, 6, 17



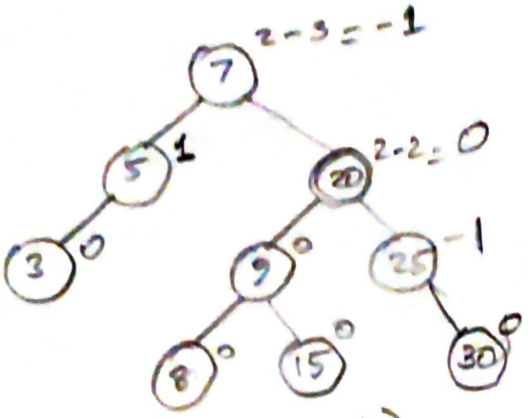




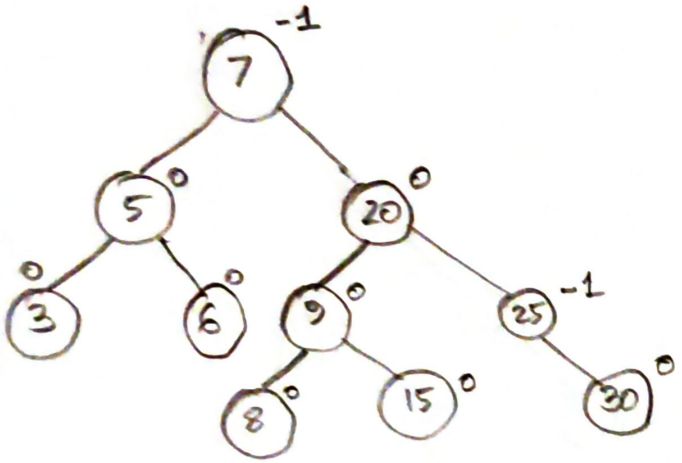




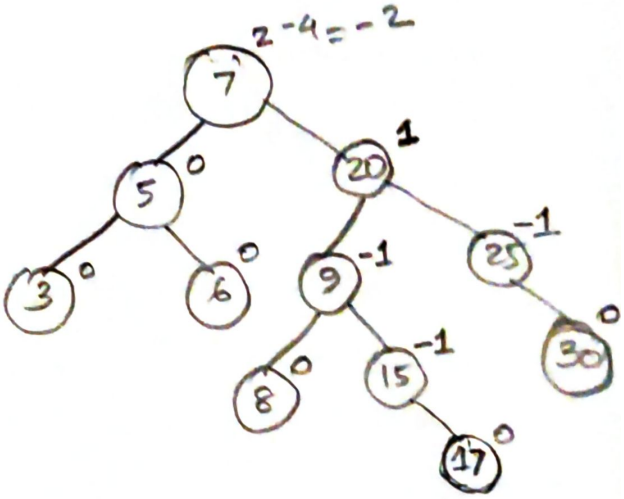
(DO Right left rotation around 9)



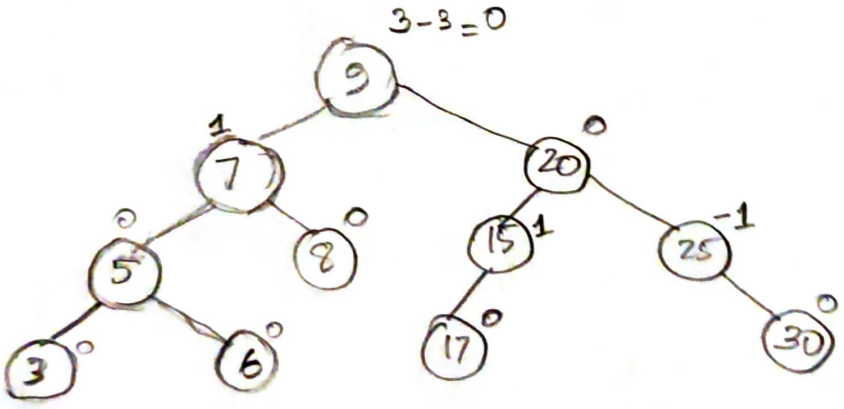
(Balanced)



(Balanced)

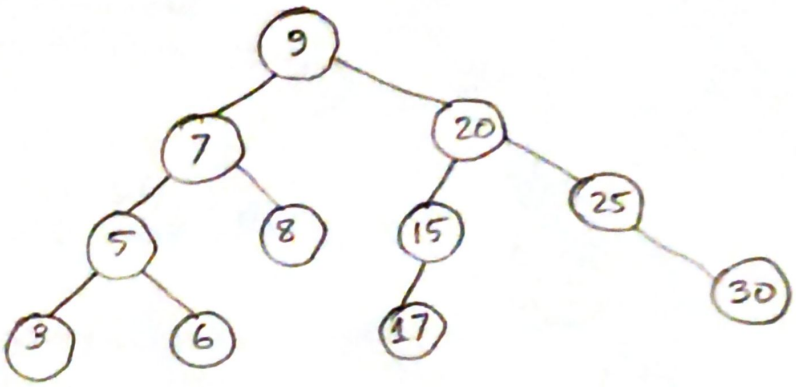


(Do Right left notation around 7)

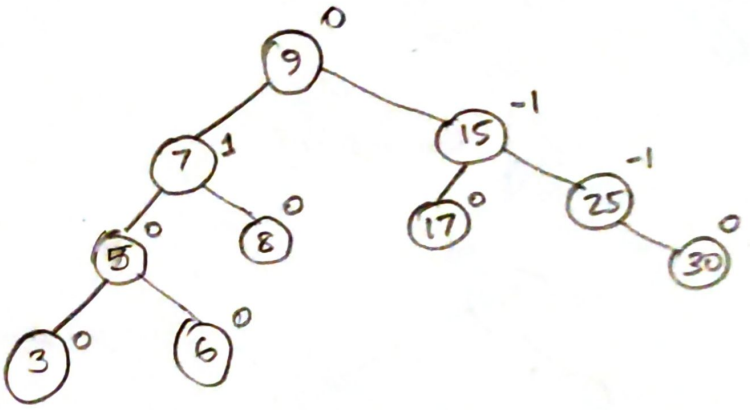


FINAL TREE

Deletion– 20, 15, 8, 25, 30, 9, 17, 5, 6, 3, 7

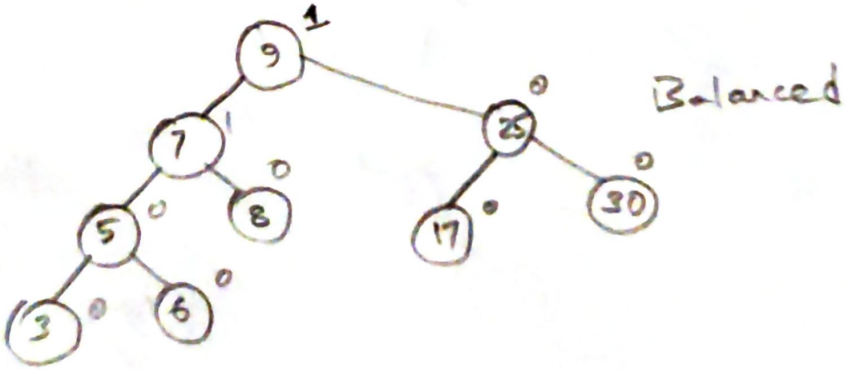


remove 20



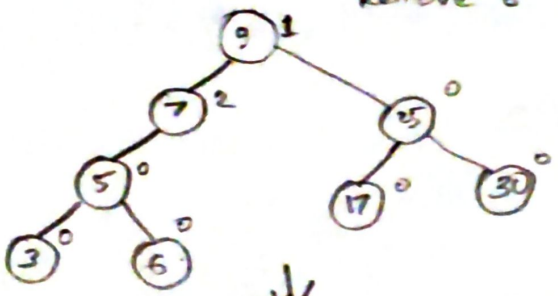
remove 15

Imbalanced so do single left around 17

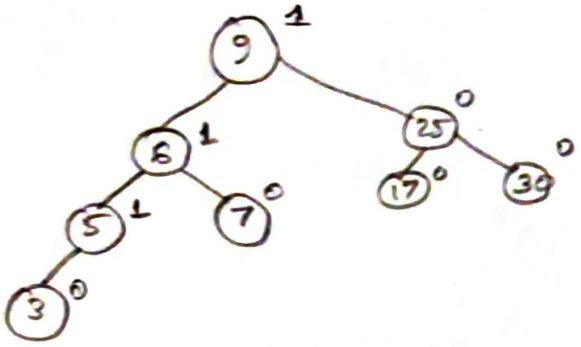


Balanced

Remove 8

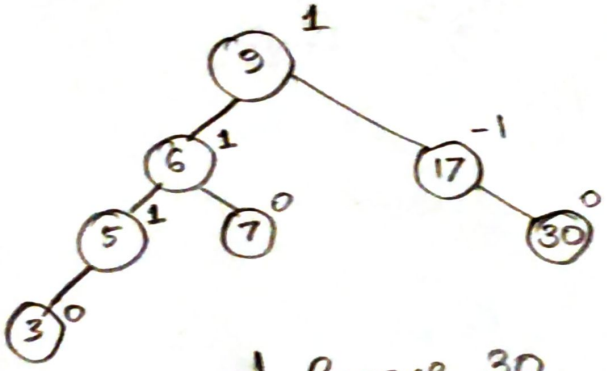


Imbalanced-do left-right rotation around 7



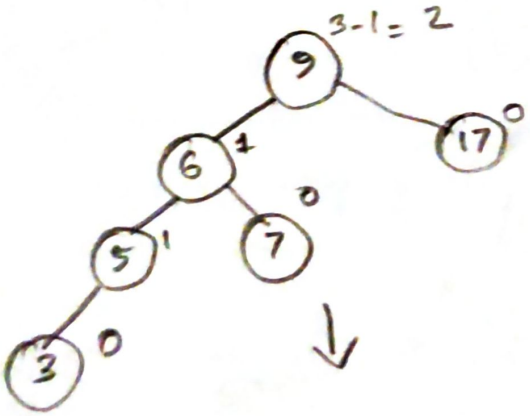
Balanced

Remove 25

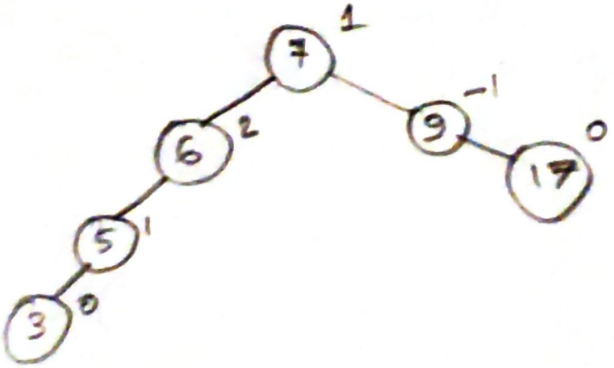


Balanced

Remove 30



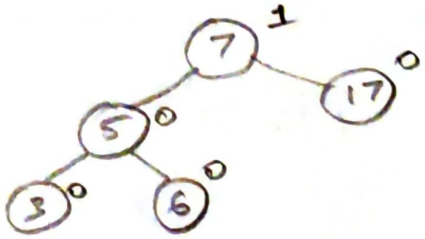
Imbalanced do left right rotation around 9.



Again imbalance, rotate single right around 6.

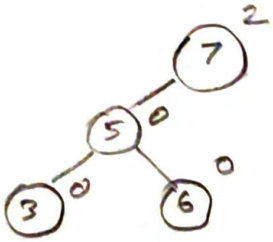
Balanced

↓ Remove 9



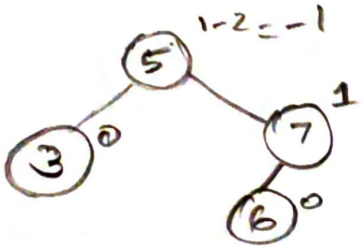
Balanced

↓ Remove 17



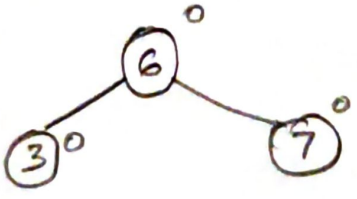
Imbalanced so do single right around 7

↓



Balanced

↓ Remove 5



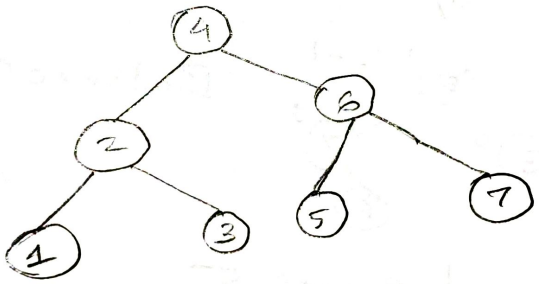
Balanced

↓ Remove 6

Remove 3 → (7) → Remove 7 and complete!

3. {1, 2, 3, 4, 5, 6, 7}

Since odd number of elements, take the median as root and build the left and right subtrees.



Hence, the order is {4, 2, 6, 1, 3, 5, 7}