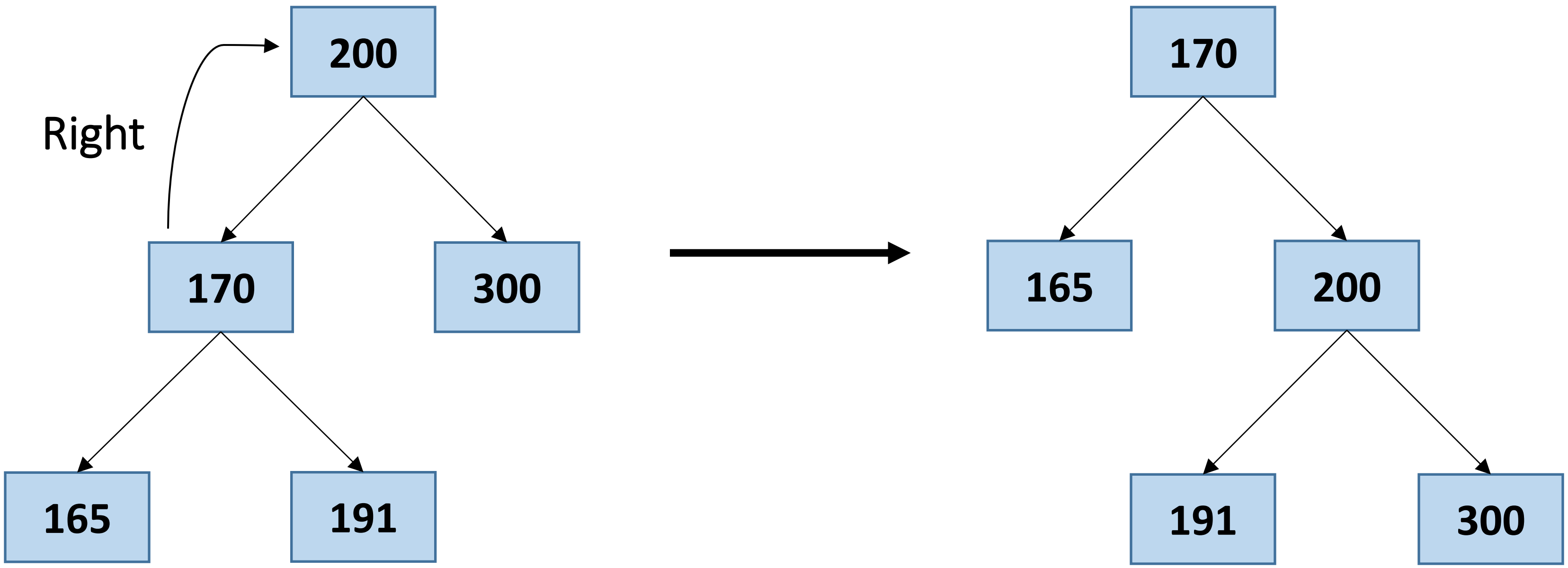
CSE 3010 – Data Structures & Algorithms

Lecture #41

What will be covered today

- Understanding rotations
- What is an AVL tree

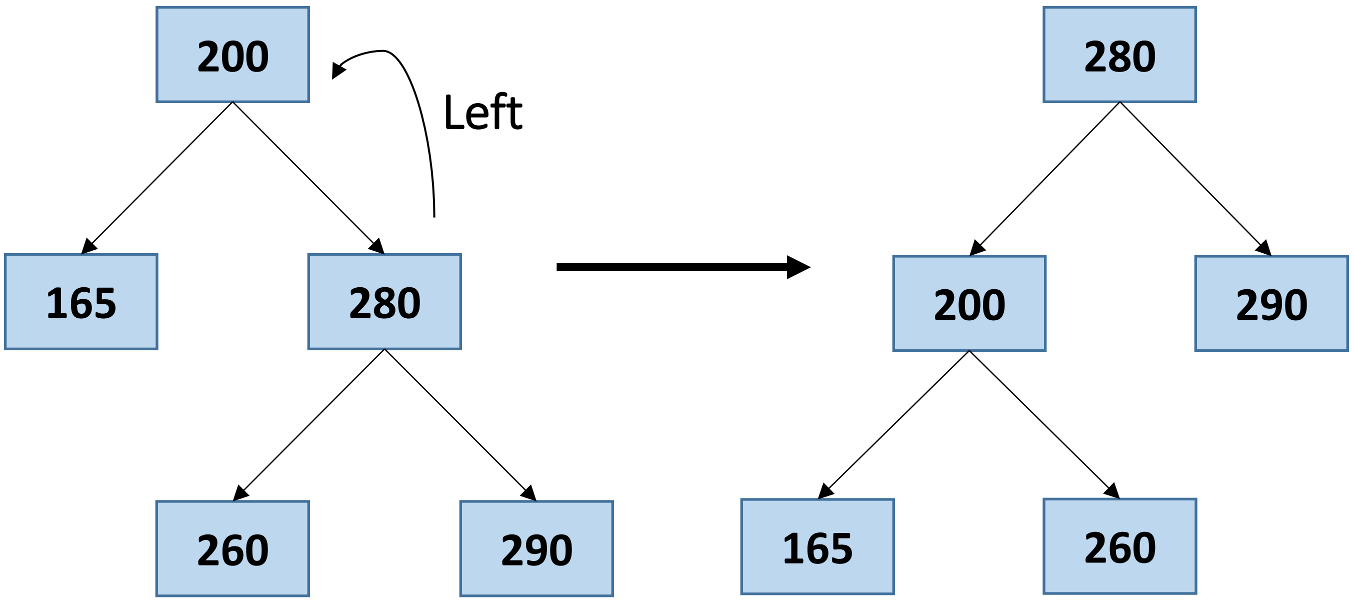
Right Rotation



Parents changed for:

- Node [170] – Becomes the root with no parent
- Node [191] – New parent is Node [200]
- Node [200] – No longer the root, new parent is node [170]

Left Rotation



Parents changed for:
- Node [280] – Becomes the root with no parent
- Node [260] – New parent is Node [200]
- Node [200] – No longer the root, new parent is node [280]

Height of a BST

- Height of a binary search tree is the length of the longest path from the root node to a leaf node

```
Height(Tree) = max(Height(Left Subtree),
Height(Right Subtree))

Height(node)
= 0, if node = NULL
= 1 + max(Height(node->left), Height(node-
>right)), if node != NULL
```

- Height of a leaf node is zero
- Height of the root node is also the height of the BST

AVL Trees

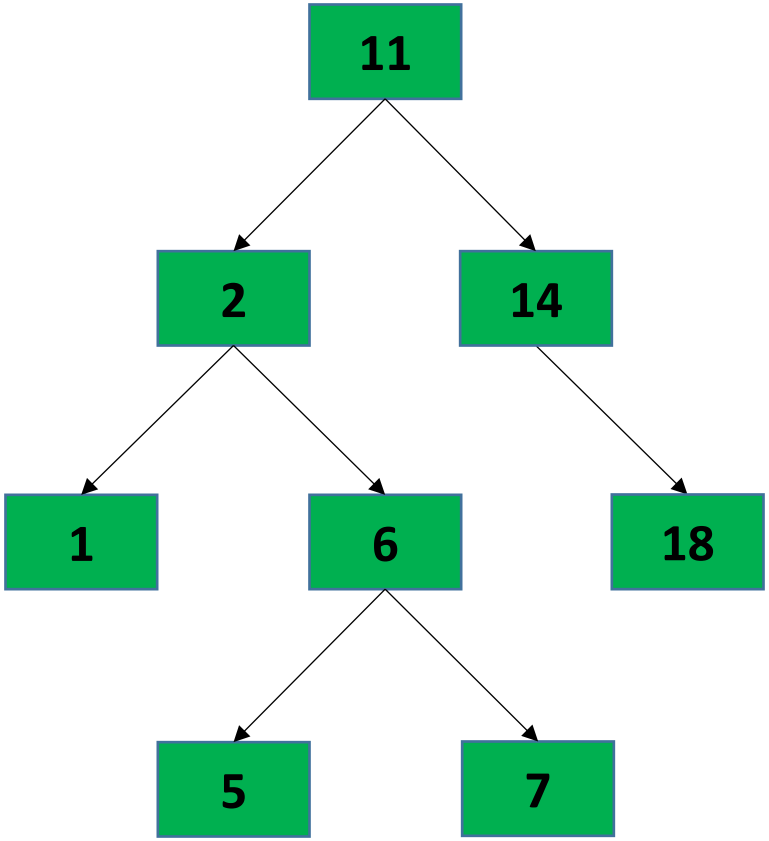
- BSTs that satisfy the additional property:

  `Balance Factor ∈ {-1, 0, 1}, where`

  `Balance Factor(node) = Height(Left Subtree) - Height(Right Subtree)`

- This property must be satisfied by all subtrees
  - Every subtree of an AVL Tree should also be an AVL Tree

Example of an AVL tree



Balance factor (BF) for every node is in the range {-1,0,1}

As examples:

BF([11]) = (3-2) = 1
BF([7]) = (0 – 0) = 0
BF([2]) = (1 – 2) = -1