

Data Structures

Minimum & Successor 1

Mostafa S. Ibrahim

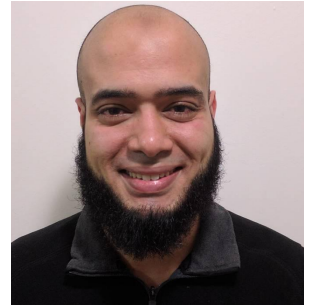
Teaching, Training and Coaching since more than a decade!

Artificial Intelligence & Computer Vision Researcher

PhD from Simon Fraser University - Canada

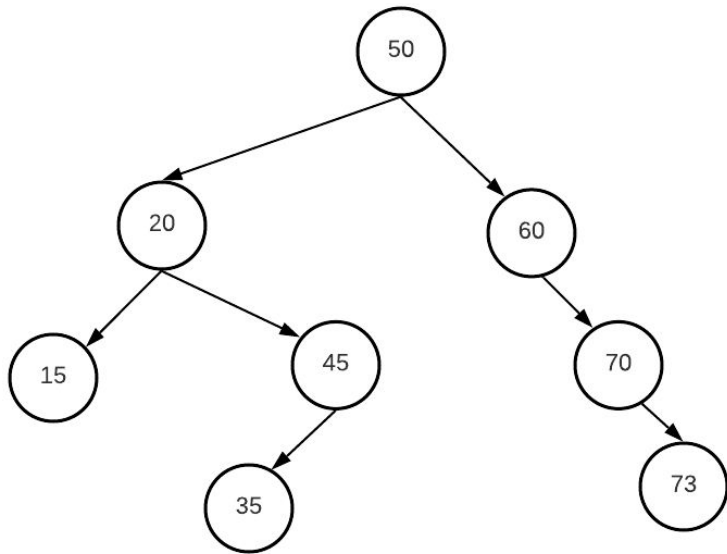
Bachelor / Msc from Cairo University - Egypt

Ex-(Software Engineer / ICPC World Finalist)



Find the minimum of BST

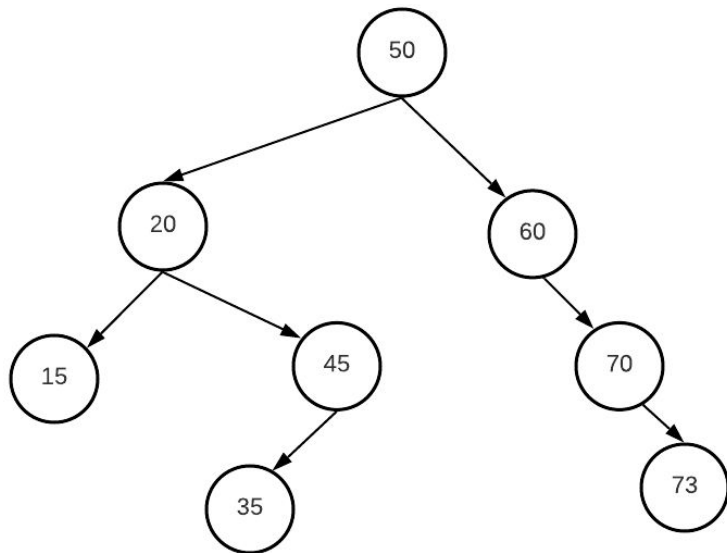
- In $O(h)$, find min of a tree
- Simply keep going to the left
tell there is no more left!
- **Why?**
- For every node, the minimum
of its 2 children is on **left**
subtree



```
int min_value() {  
    BinarySearchTree* cur = this;  
    while (cur && cur->left)  
        cur = cur->left;  
    return cur->data;  
}
```

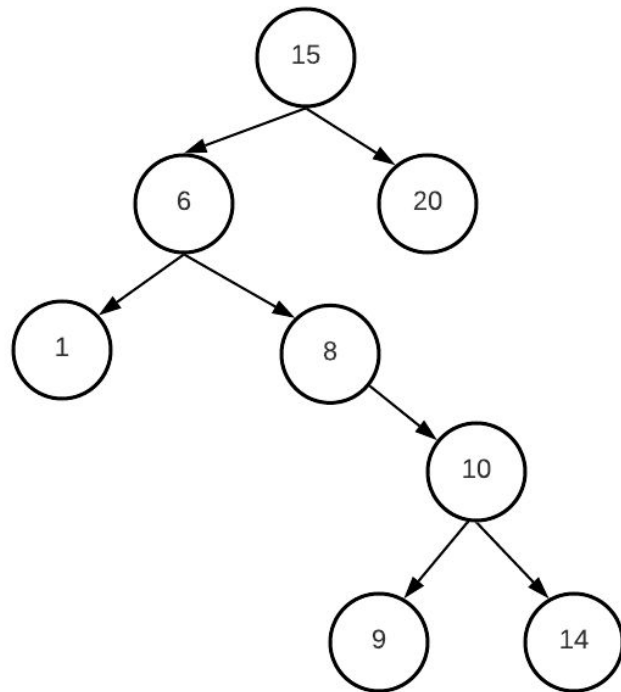
Observations

- The minimum node will **never** have a left child
 - Otherwise, it is not minimum.
 - Similarly, max node don't have right child
- In a chain of left nodes, any node is smaller than all previous values [decreasing]
 - Chain: $[50 \Rightarrow 20 \Rightarrow 15]$
 - Chain: $[45 \Rightarrow 35]$
- Similarly for chain of right nodes
 - Chain: $[50 \Rightarrow 60 \Rightarrow 70 \Rightarrow 73]$
 - E.g. 73 is greater than all previous values



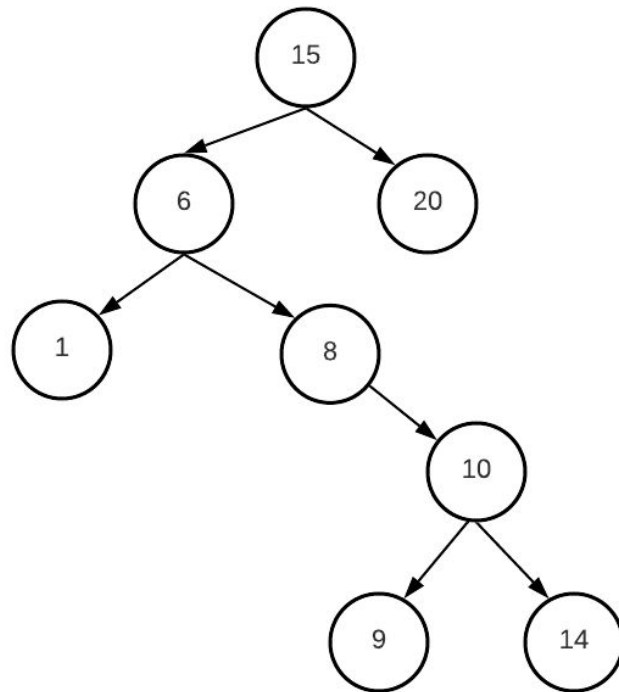
Inorder **Successor** in Binary Search Tree

- Given node x , find node y that is the smallest $y > x$ [in $O(h)$]
 - Inorder: 1 6 8 9 10 14 15 20, but $O(n)$
- Let's first find the node
- Where are greater values than me?
 - Either on my right tree?!
 - Or maybe somewhere between me and parent?!
 - Or maybe right of some parent?!
- But we only seek the smallest $y > x$?
 - There are only 2 cases then!
 - Think of your right child more!



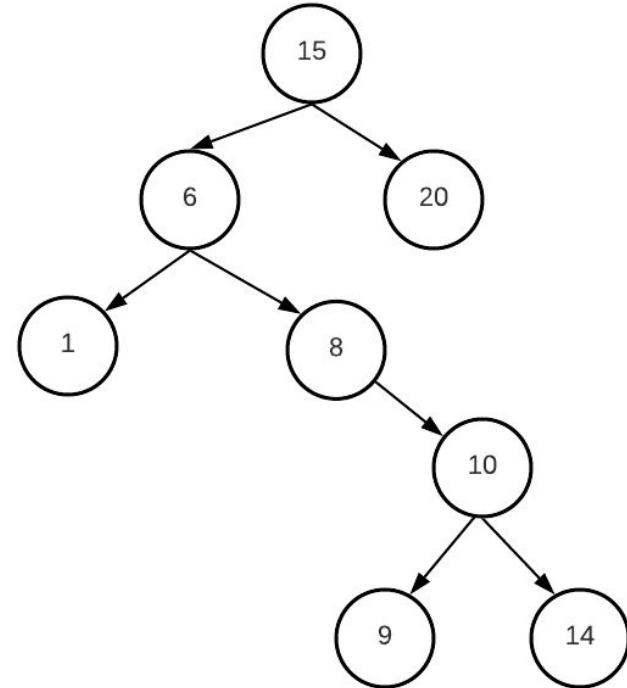
Inorder **Successor** in Binary Search Tree

- Consider $x = \text{Node}(8)$
- It has a right subtree
 - We have values: $[9, 10, 14]$, all must be > 8
 - Answer is $\min(\text{right}) = 9$



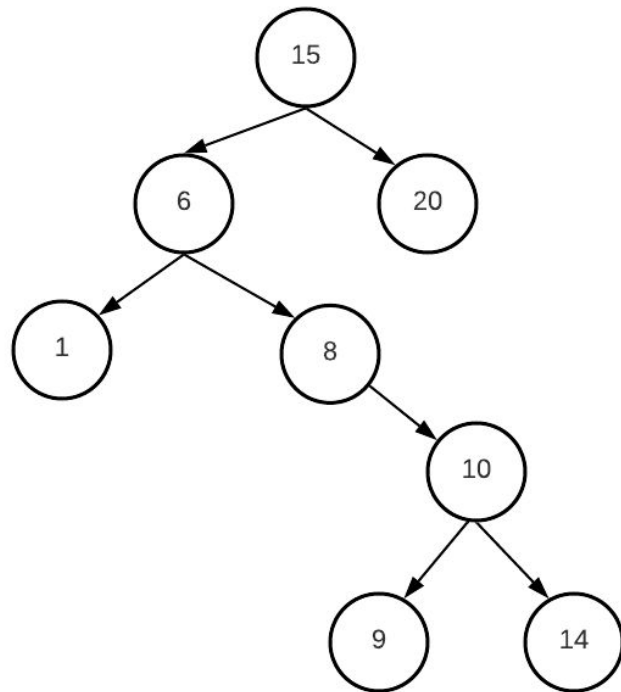
Inorder **Successor** in Binary Search Tree

- Consider $x = \text{Node}(14)$
- At 14: Is right child? Yes \Rightarrow go up
- At 10: Is right child? Yes \Rightarrow go up
- At 8: Is right child? Yes \Rightarrow go up
- At 6: Is right child? No
- What is parent? 15 \Rightarrow Successor



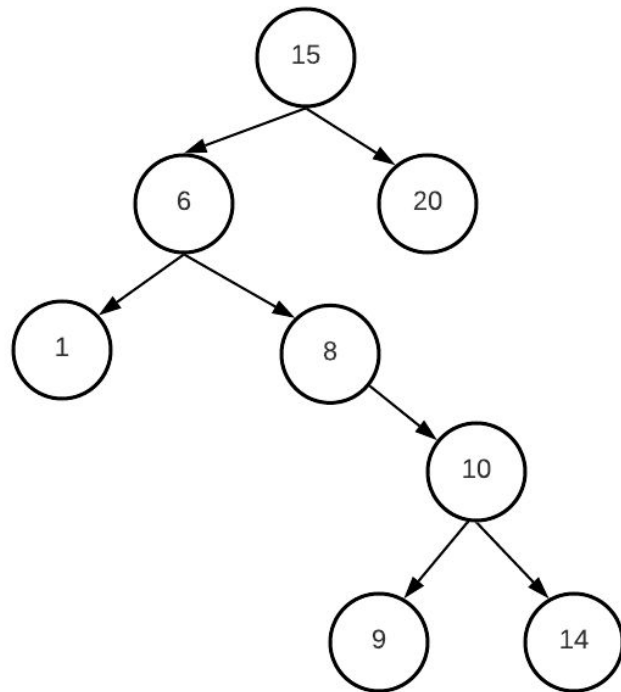
Inorder **Successor** in Binary Search Tree

- Consider $x = \text{Node}(9)$ - No right Child
 - Go up procedure
- At 9: Is right child? No
- What is parent? 10 \Rightarrow Successor



Inorder **Successor** in Binary Search Tree

- Consider $x = \text{Node}(20)$ - No right Child
 - Go up procedure
- At 20: Is right child? Yes \Rightarrow go up
- At 15: Root!
 - 20 is on its right, so 20 is bigger!
 - But it is the root, no more up
- No Successor for 20
 - Happens only for the max value in the tree



“Acquire knowledge and impart it to the people.”

“Seek knowledge from the Cradle to the Grave.”