

# Binary Search Trees: Delete

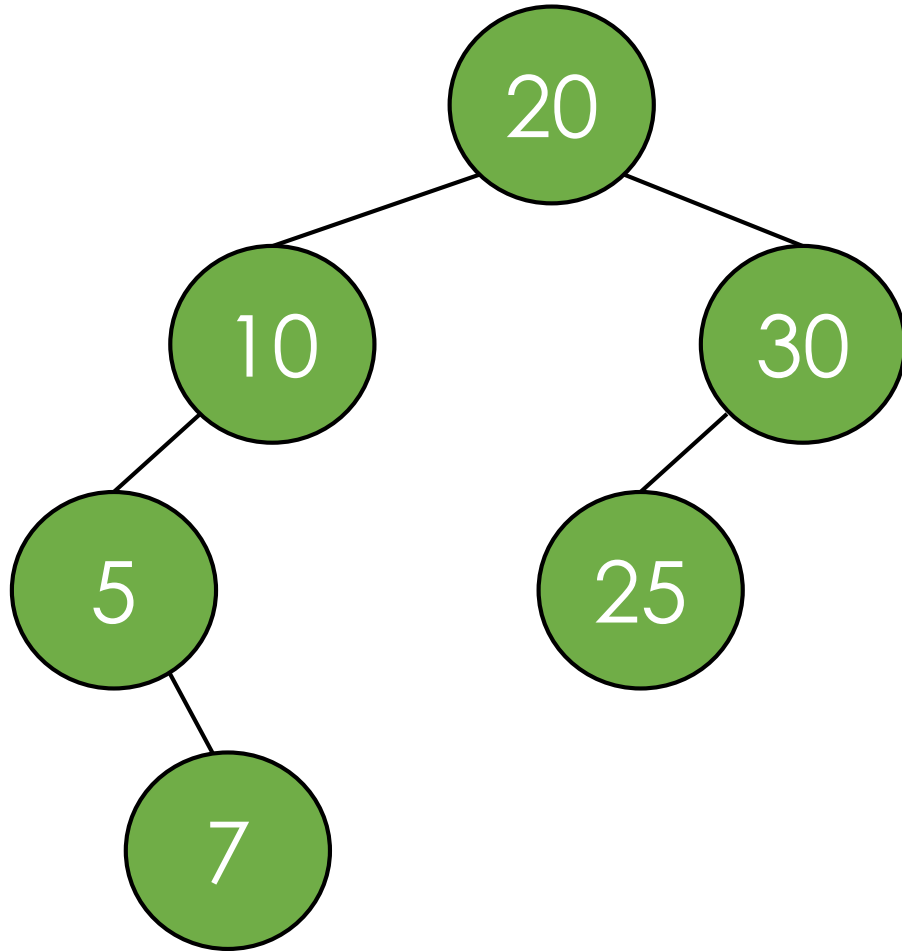


This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/)  
by Christine Alvarado, Mia Minnes, and Leo Porter, 2015.

**By the end of this video you will be able to...**

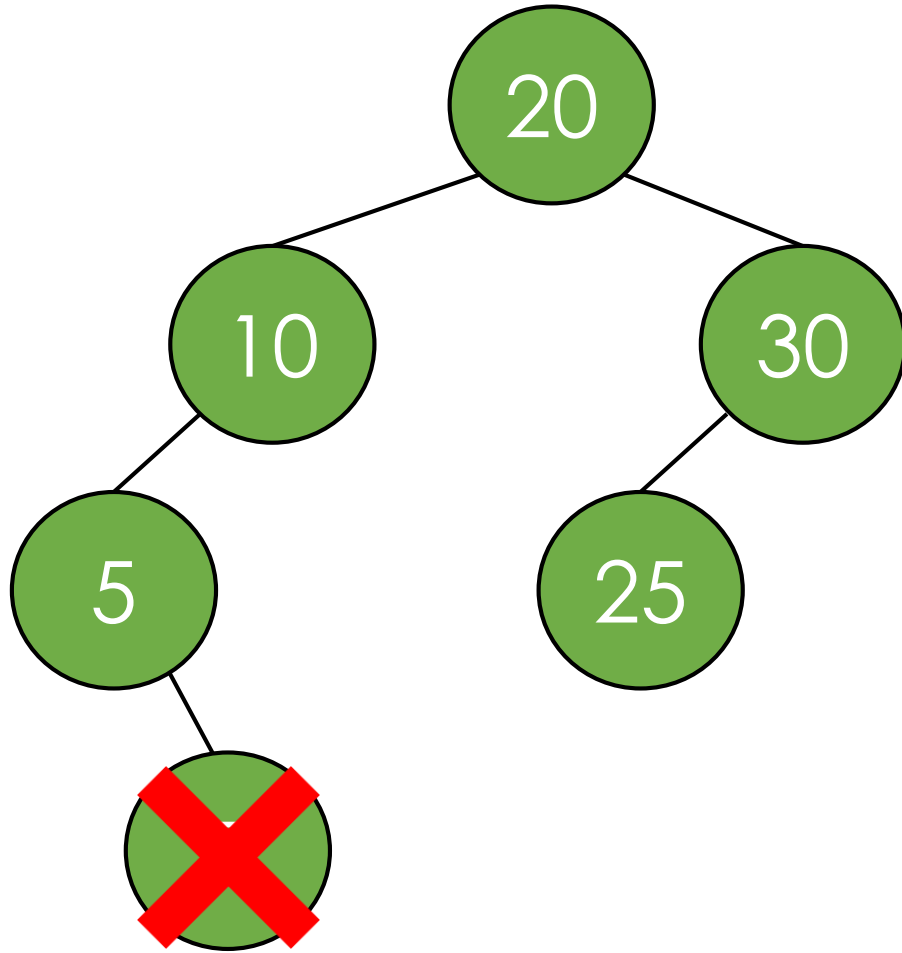
- Delete an item from a Binary Search Tree

# BST - Deletion



**Delete 7**

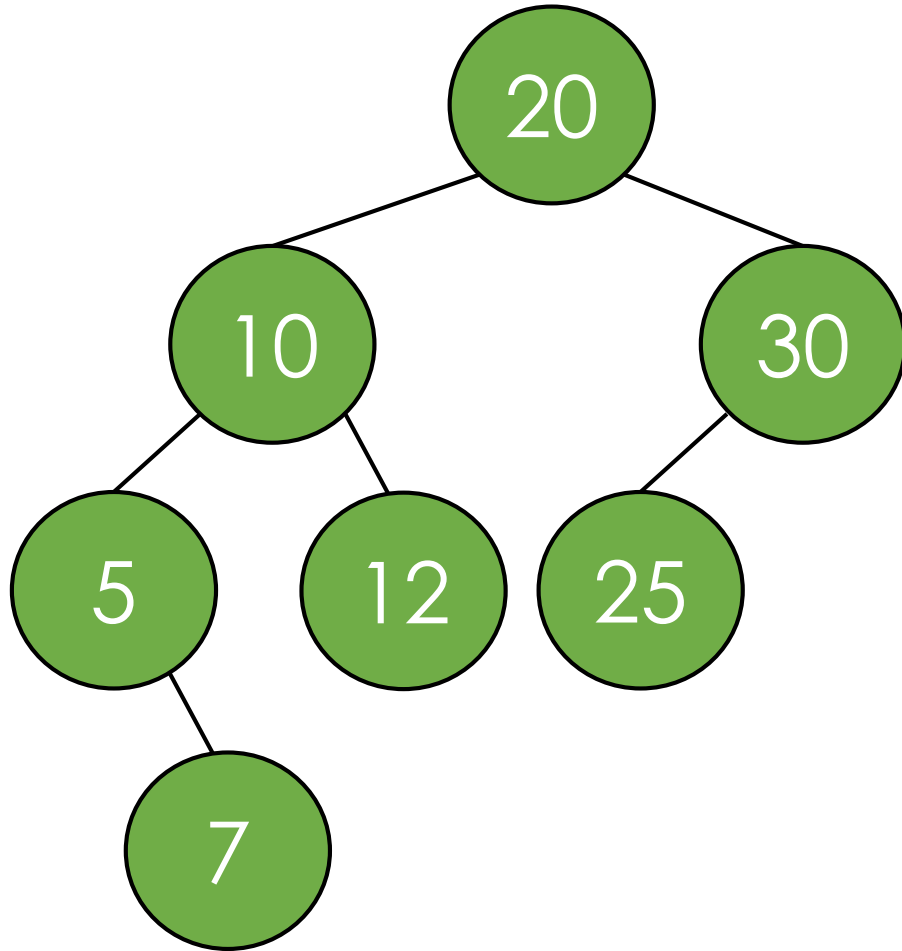
# BST - Deletion



**Delete 7**

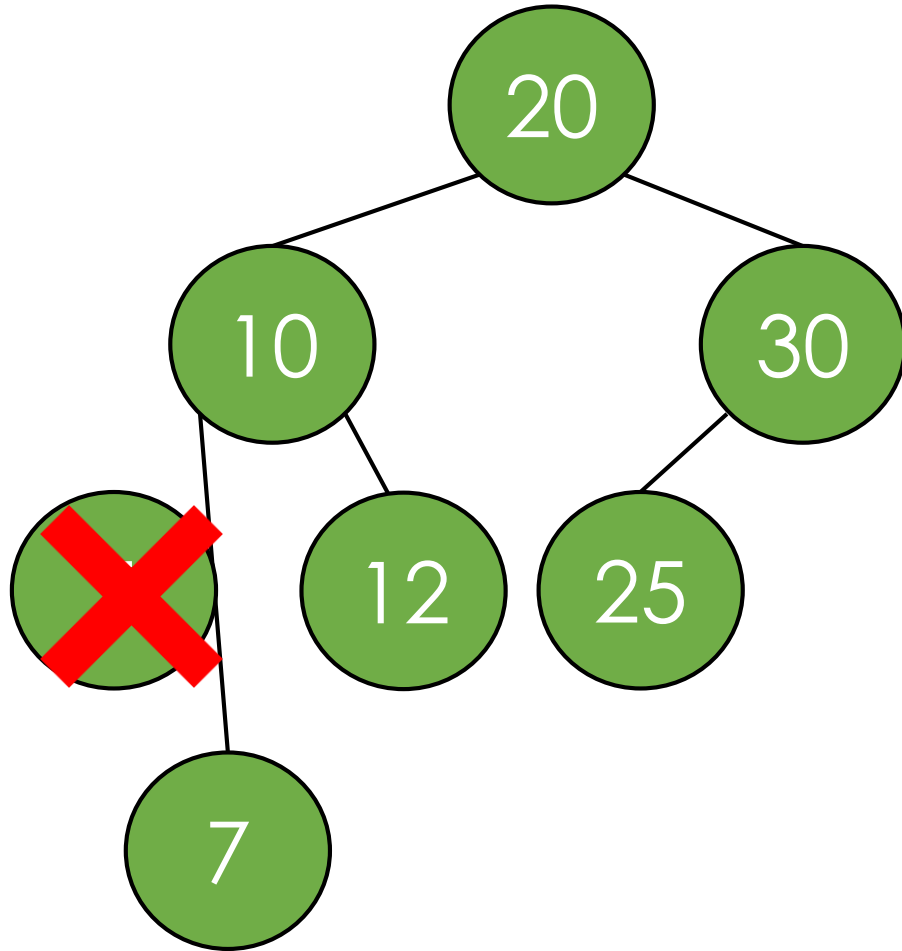
**If leaf node: Delete  
parent's link 7**

# BST - Deletion



**Delete 5**

# BST - Deletion

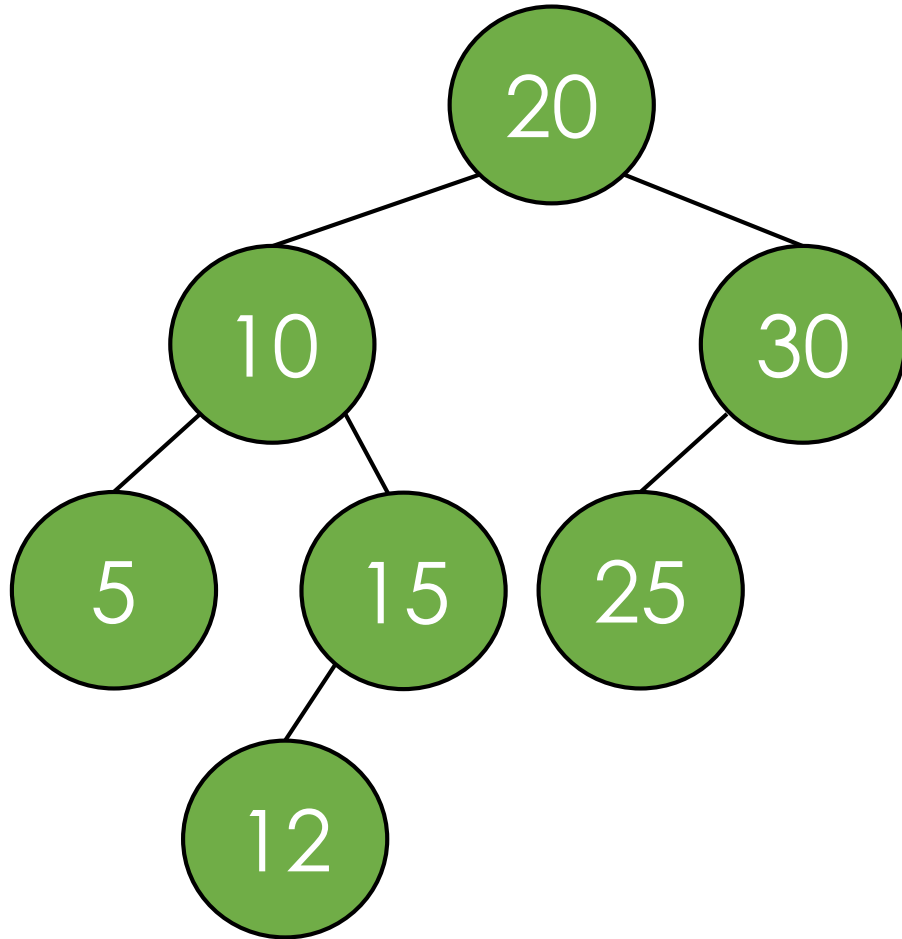


**Delete 5**

**If only one child,  
hoist child**

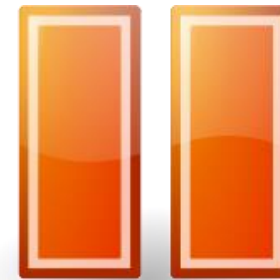
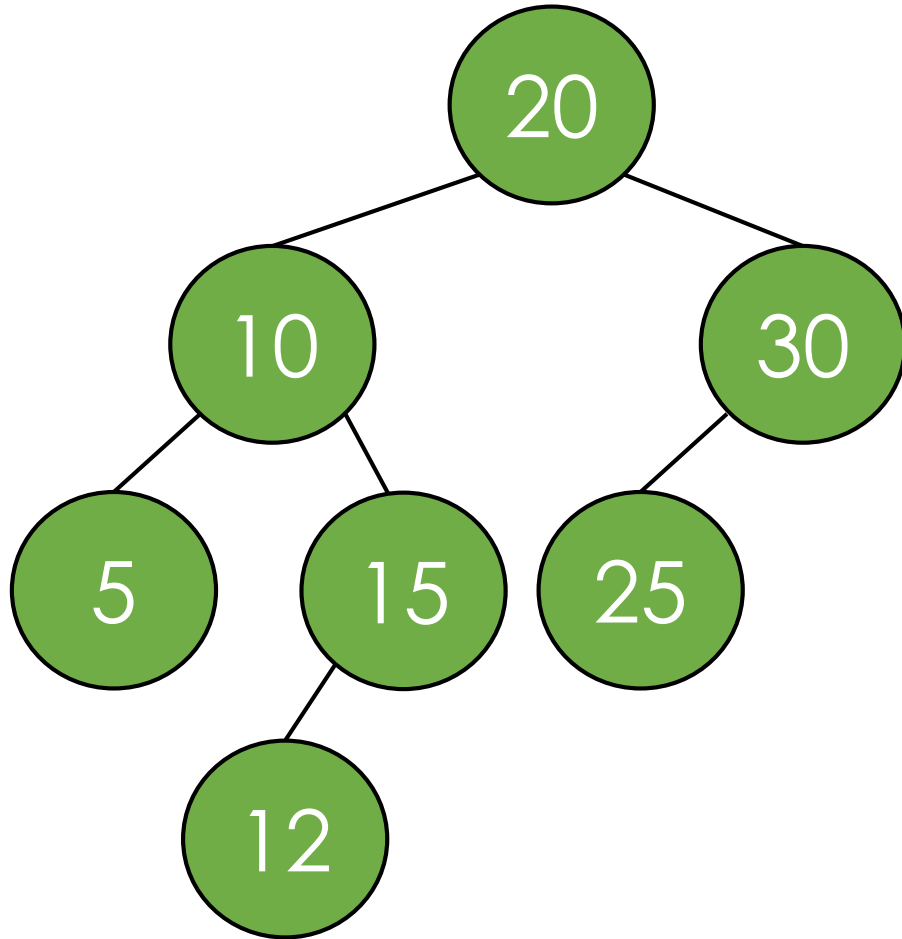
# BST - Deletion

**Delete 10**



# BST - Deletion

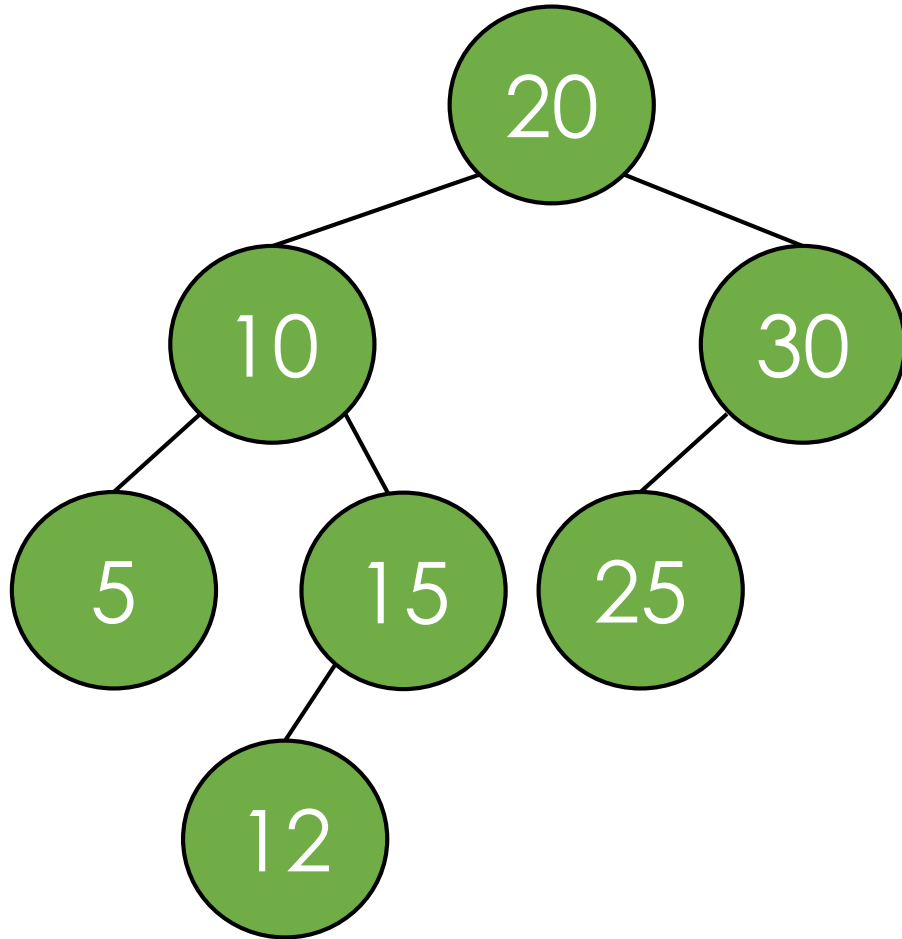
**Delete 10**



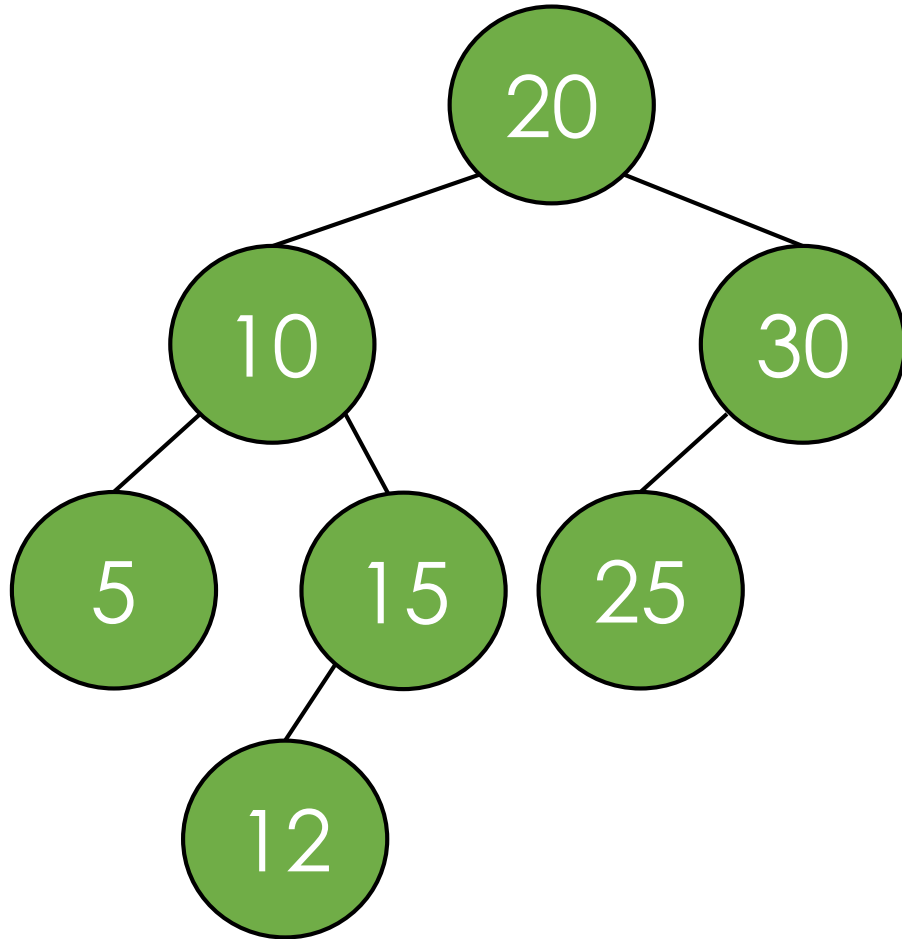


# BST - Deletion

**Delete 10**



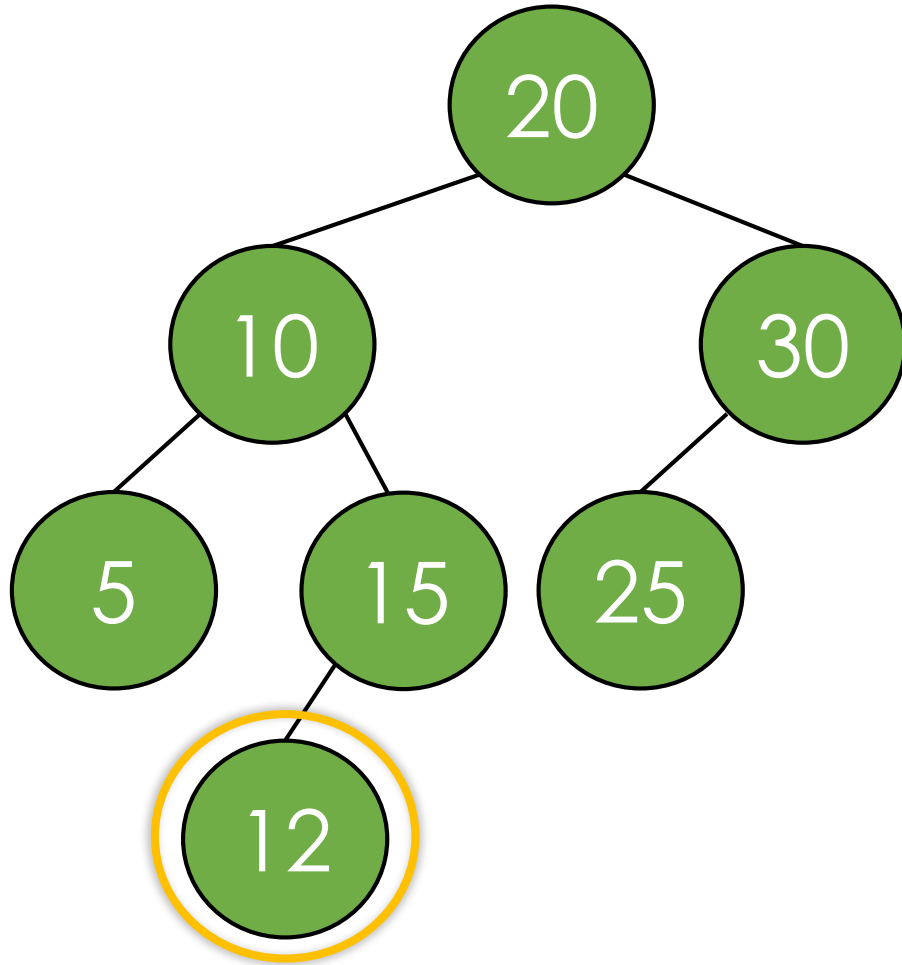
# BST - Deletion



**Delete 10**

**When a deleted node has two children, this gets tricky.**

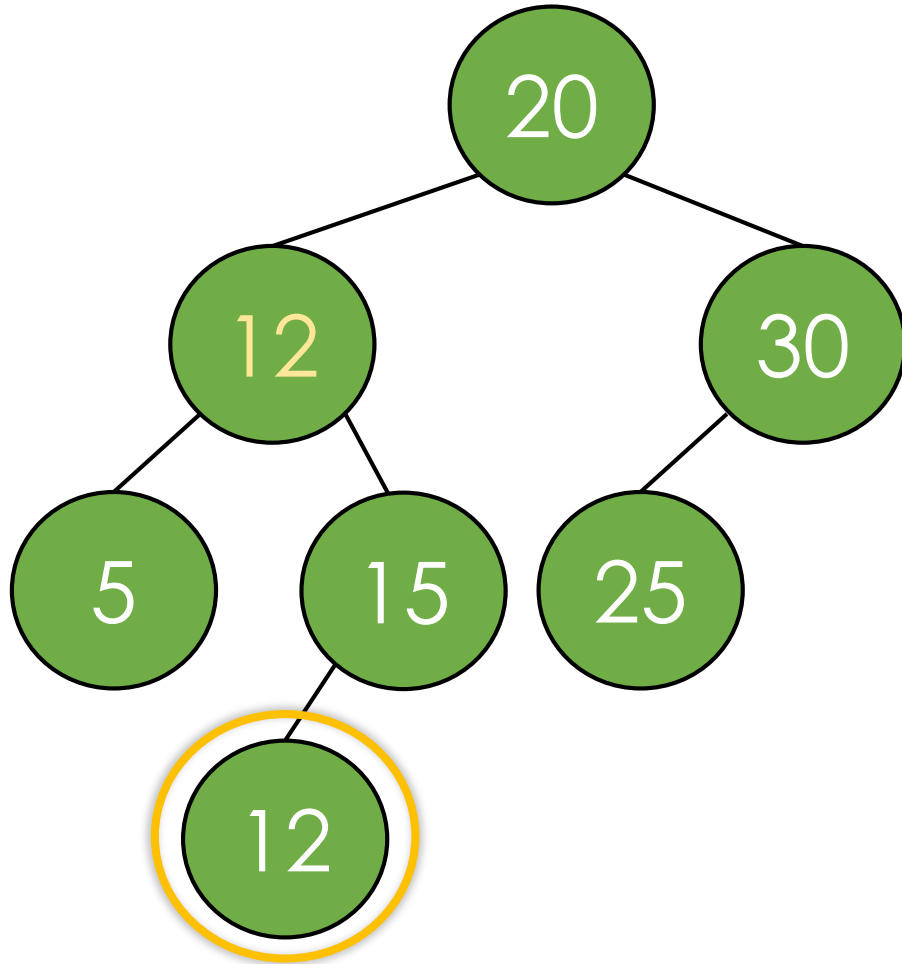
# BST - Deletion



**Delete 10**

**Find smallest value in  
right subtree**

# BST - Deletion

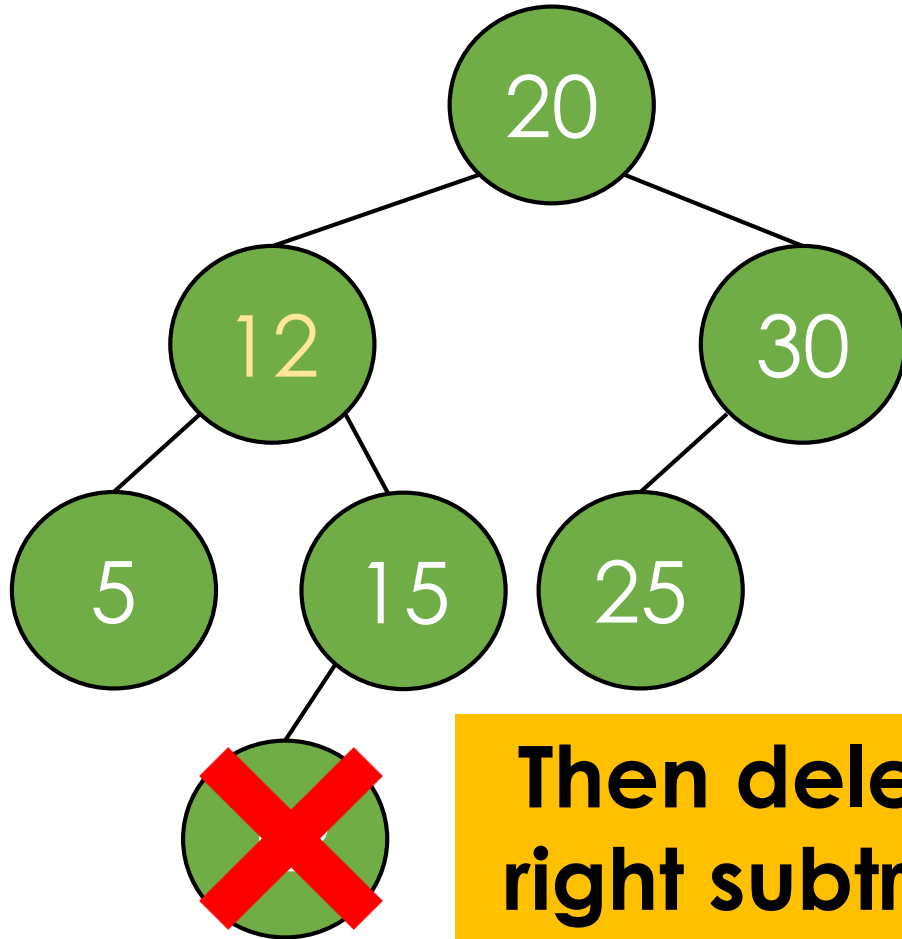


**Delete 10**

**Find smallest value in  
right subtree**

**Replace deleted  
element with smallest  
right subtree value**

# BST - Deletion



**Then delete  
right subtree  
duplicate  
(12)**

**Delete 10**

**Find smallest value in  
right subtree**

**Replace deleted  
element with smallest  
right subtree value**

## In this lesson you have:

- Learned how to create Trees
- Explored different traversal options
- Worked Closely with Binary Search Trees
  - Search, Insert, and Delete