

CSE 12 – Basic Data Structures and Object-Oriented Design

Lecture 18

Greg Miranda, Spring 2021

Announcements

- Quiz 18 due Wednesday @ 12pm
- PA6 due Wednesday @ 11:59pm (open)
- Survey 7 due Friday @ 11:59pm

Topics

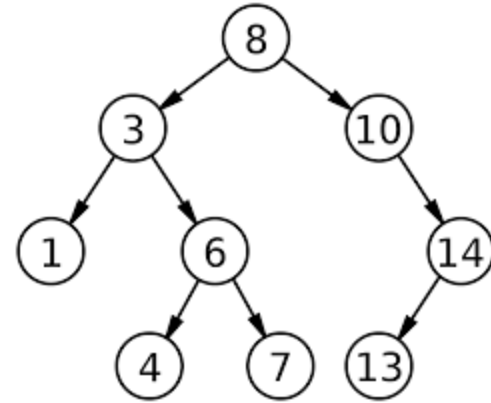
- Binary Search Trees
- Questions on Lecture 18?

```
class Node<K,V> {  
    K key;  
    V value;  
    Node<K,V> left;  
    Node<K,V> right;  
    public Node(K key, V value,  
                Node<K,V> left,  
                Node<K,V> right) {  
        this.key = key;  
        this.value = value;  
        this.left = left;  
        this.right = right;  
    }  
}
```

```
class BST<K, V> {  
    Node<K, V> root;  
    BST() (this.root = null);  
    BST(Node<K, V> root) { this.root = root; }  
  
    V get(Node<K, V> node, K key) {  
        if (node == null) { //throw error }  
        if (node.key.equals(key)) {  
            return node.value;  
        }  
        if (node.key > key) {  
            return get(node.left, key);  
        }  
        else {  
            return get(node.right, key);  
        }  
    }  
  
    V get(Key key) {  
        return this.get(root, key);  
    }  
}
```

Binary Search Tree

- Assume the key and value are identical for this example
- Trace the path for get(4)
 - How many nodes does it touch?
- Trace the path for get(2)
 - How many nodes does it touch?
 - What happens when the nodes isn't found?



Binary Search Tree

- Assume the key and value are identical for this example
- Trace the path for `get(40)`
 - How many nodes does it touch?
- Trace the path for `get(4)`
 - How many nodes does it touch?

