CSE12 - Lecture 19

Friday, May 17, 2024 10:00 AM

class Student {
String name;

new SortbyName()

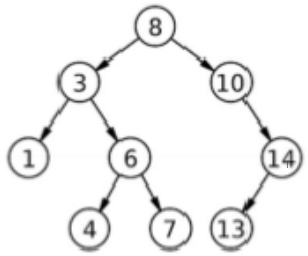
Lecture 19 Traversal

What order does printAllElement() traverse the tree?

```
void printAllElements(Node<K, N> n) {
  if (n == null ) return;
  System.out.println(n.key);
  printAllElements(n.left);
  printAllElements(n.right);
}

void printAllElement() {
  printAllElements(this.root);
}
```

What's the post, pre, in-order traversal of this tree?



Traversal

① pre-order → 8 3 1 6 4 7 10 14 13

③ post-order → 1 4 7 6 3 13 14 10 8

② in-order → 1 3 4 6 7 8 10 13 14

```
class BSTMap<K,V> implements OrderedDefaultMap<K,V>{
  Node<K, V> root;
  int size;
  Comparator<K> comparator;
  ...
  Node<K, V> set(Node<K, V> node, K key, V value) {
    if (node == null) {
      this.size += 1;
      return new Node<K, V>(key, value, null, null);
    }
    int comp = this.comparator.compare(node.key, key);
    if (comp < 0) {
      node.right = this.set(node.right, key, value);
      return node;
    } else if (comp > 0) {
      node.left = this.set(node.left, key, value);
      return node;
    } else {
      node.value = value;
      return node;
    }
  }

  @Override
  public void set(K key, V value) {
    if (key == null) {
      throw new IllegalArgumentException();
    }
    this.root = this.set(this.root, key, value);
  }
}
```

Use the picture on the left and assume the key and value are identical:

set("5", 5);
set("11", 11);
set("15", 15);
set("12", 12);

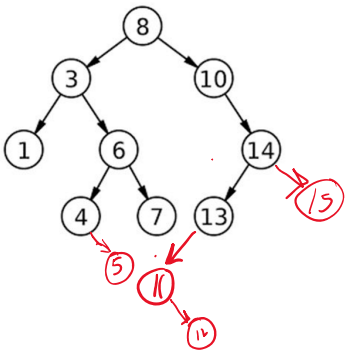
What is the picture after calling the above set() methods?

Name: _____ PID: _____ Code: 4327

```
// Class 3
// Helper class implementing Comparator interface
class Sortbyname implements Comparator<Student> {

    // Method
    // Sorting in ascending order of name
    public int compare(Student a, Student b)
    {

        return a.name.compareTo(b.name);
    }
}
```



What is the run-time for a Binary Tree

set()

Worst Case $\Theta(N)$

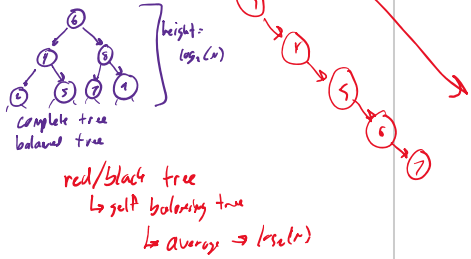
What conditions make up the worst case for set()?

sorted or reverse sorted

Best Case: $\log_2(N)$

What conditions make up the best case for set()?

root nodes are median values



red/black tree
↳ self balancing tree
↳ average → $\log(n)$

get()

Worst Case $\Theta(N)$

What conditions make up the worst case for get()?

sorted / reverse sorted → largest / last element or not found

Best Case: $\Theta(1)$

What conditions make up the best case for get()?

root node has the key we are looking for

printAllElements()

Worst Case → $\Theta(N)$

Best Case: →

What conditions make up the best case for set()?

- N- Set
- N- write
- print