

11/11/20

AKSHAY MITTAR
IBM18CS010

ADS WRITE UP

classmate

Date

Page

Page 1

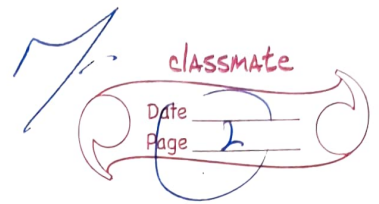
B-Tree

Insertion functions:

```
void insert(int k) {  
    if (!root) {  
        root = new Node(t, leaf=true);  
        root->keys[0] = k;  
        root->n = 1;  
    } else {  
        if (root->n == 2*t-1) {  
            Node *s = new Node(t, leaf=false);  
            s->[0] = root;  
            s->splitChild(0, root);  
            int i = 0;  
            if (s->keys[0] < k) i++;  
            s->[i] = insertNonFull(k);  
            root = s;  
        } else root->insertNonFull(k);  
    }  
}  
  
void insertNonFull(int k) {  
    int i = n-1;  
    if (leaf == true) {  
        while (i >= 0 && keys[i] > k) {  
            keys[i+1] = keys[i];  
            i--;  
        }  
    }  
}
```

11/11/20

AKSHAY MITTUR
IBM18CS010



```
keys[i+1] = k;  
n += 1;  
} else {  
    while(i >= 0 && keys[i] > k) i--;  
    if([i+1] → n == 2*t - 1) {  
        splitChild(i+1, [i+1]);  
        if(keys[i+1] < k) i++;  
    }  
    [i+1] → insertNonFull(k);  
}
```

splitChild Function Splits the Node and promotes 1 key to parent while making the other ones children.

