

2-3 Trees

Insertion -

```
if (root == NULL)
```

```
{  
    root = new Treenode(true);
```

```
    root->key[0] = K
```

```
    root->n = 1
```

```
}
```

```
else
```

```
{
```

```
    if (root->n == 3)
```

```
{
```

```
        Treenode s = new Treenode(false)
```

```
        s->child[0] = root
```

```
        s->splitchild(0, root)
```

```
        if s->key[0] < K
```

```
            i++
```

```
        s->child[i] = insertNotfull(K)
```

```
        root = s
```

```
    }
```

```
else
```

```
    root->insertNotfull(K)
```

```
}
```

Solution

```

int id = findKey(x)
if (id < n && Key[id] == k)
{
    if (leaf)
        removefromleaf(id);
    else
        removefromNonleaf(id)
}
else
{
    if (leaf)
        Key not exist
        flag = id == n ? true : false
        if (child[id] → n < 2
            fill(id)
            if (flag && id > n)
                child[id - 1] → remove(k)
            else
                child[id] → remove(k)
        }
    return;
}

```