2-11-20

ADS LAB
2-3 trees

1. Abhiram G
1BM18CS127

Insert :

```
void insert (int k) {
    if ( !root) {
        root = new TreeNode (true);
        root -> keys [0] = k;
        root -> n = 1;
    } else {
        if (root -> n == 3) {
        Treenode * s = new Treenode(false);
        s -> child [0] = root;
        s -> split (child (0, root);
        int i = 0;
        if( s -> keys [0] < k) ++ i;
        s -> child [i] -> insertNonFull(k);
        root = s;
    } else
        root -> insert Non full (k);
}
```

Deletion :

```
void delete (int k) {
    int id = find (k);
    if (id < n && key s [id] == k) {
        if (leaf) remove fromLeaf (id);
        else removefrom non leaf (id);
    } else {
        if (leaf) {
        cout << " doest exist " << endl;
    }
    bool flag = ((id == n) ? true : false);
```

```
    if (child[id]->n<t) fill(id);
    if (flag && id>n)
        child[id-1]->remove(k);
    else child[id]->remove(k);
  }
  return;
}
```

Auxillary funct:

removeFromLeaf => shifts elements to left upon deleting a node

removeFromNonLeaf => merges node after deletion

splitChild => splits a node

insertNonFull => inserts key into Node