ADS- LAB
(Write-up)
2-3 Trees

Abhishikat Kumar Sa
1BM18CS004
5A

Insertion:

```
void insert(int k)
{ if(!root)
   { root = new TreeNode(true);
     root -> keys[0] = k;
     root -> n = 1;
   }
 else {
   if(roo -> 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 -> insertNonFull(k);
    }
   }
}
```

Deletion

```
void delete(int k)
{ int index = find(k);
  if (index <n && keys[index] = = k)
   { if(leaf)   remove from leaf (index);
     else       remove from Non-leaf (index);
   }
```

Abhishikat Kuman Soni
1BM18CS004
5A

```
else
{ if(leaf)
  { cout<<"Not Exist"<<endl;
      return;
  }
 bool flag = ((index==n)? true:false);
 if (child[index]->n < 2)    fill(index);
 if (flag  && index>n)
          child[index-1] -> remove(k);
 else
      child[index] -> remove(k);
  }
  return;
}
```

Extra functions

// insertNotfull => insert key into node

// splitChild => Split node => children node

// RemovefromNonleaf => Merges node after deletion

// Remove from Leaf => Shift element to leftside after deletion.

2