

```
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->insertNonFull (k);
        }
    }
}
```

```
void delete (int k)
{
    int idx = find (k);
    if (idx < n && keys [idx] == k)
    {
        if (leaf) removeFromLeaf (idx);
        else removefromNonLeaf (idx);
    }
    else
    {
        if (leaf)
        {
            cout << "doesn't exist" << endl;
            return;
        }
    }
}
```

Date _____
Page _____

```

bool flag = (idx == n) ? true : false;
if (child[idx] → n < 2) fill(idx);
if (flag && idx > n)
    child[idx-1] → remove(k);
else child[idx] → remove(k);
}
return;
}

```