Insertion in B-Tree

```
void insert_btree(int k)
{
    if (root == NULL)
    {
        root = new BNode(t, true);
        root → keys[0] = k;
        root → n = 1;
    }
    else {
        if (root → n == 2*t - 1)
        {
            BNode *s = new BNode(t, false);
            s → C[0] = root;
            s → splitChild(0, root);
            int i = 0;
            if (s → keys[0] < k)
                i++;
            s → C[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--;
        }
        keys[i+1] = k;
        n = n+1;
    }
    else
    {
        while (i>=0 && keys[i] >k)
            i--;
        if (c[i+1]->n == 2*t -1)
        {
            splitChild(i+1, C[i+1]);
            if (keys[i+1] <k)
                i++;
        }
        c[i+1]-> insertNonFull(k);
    }
}
```