

توضیحات سوال ۲ پروژه پنج

علیرضا سعیدنیا – ۴۰۰۱۰۸۳۳

میدانیم در هر باینری سرچ تری عنصر سمت چپ از عنصر پرنت کوچکتر است و عنصر سمت راست از عنصر پرنت بزرگتر است برای درج همین را فرض میکنیم. در سوال یک باینری سرچ تری فرضی قرار داده ام و میخواهم عنصر سمت راست ریشه ان را که برابر هفتاد است حذف کنم

```
static Node insert(Node node, int key)
{
    if (node == null)
        return newNode(key);

    if (key < node.key)
        node.left = insert(node.left, key);
    else
        node.right = insert(node.right, key);

    return node;
}
```

با دستور پرینت هم درخت را پرینت میکنیم

```
public static void print(String prefix, Node n, boolean isLeft) {
    if (n != null) {
        print(prefix + "      ", n.right, isLeft: false);
        System.out.println(prefix + ("|-- ") + n.key);
        print(prefix + "      ", n.left, isLeft: true);
    }
}
```

```
      |-- 80
    |-- 70
      |-- 60
|-- 50
      |-- 40
    |-- 30
      |-- 20
```

حال نوبت به حذف میرسد ، برای حذف اگر ریشه ای که میفرستیم نال بود که هیچی نال برمیگرداند اگر بزرگتر بود که در این مساله بزرگتر است ان را به عنوان روت بعدی در نظر میگیریم ، در حال بعدی حالا روت برابر ۷۰ میشود ، ۷۰ نال نیست ، کوچکتر از ۷۰ نیست ، بزرگتر از ۷۰ نیست پس وارد خط های بعدی میشود

```
if (root == null)
    return root;

if (root.key > k) {
    root.left = deleteNode(root.left, k);
    return root;
}
else if (root.key < k) {
    root.right = deleteNode(root.right, k);
    return root;
}
```

در عکس پایین چک میکند که ایا تک فرزند است یا جفت فرزندی

```
if (root.left == null) {
    Node temp = root.right;
    return temp;
}
else if (root.right == null) {
    Node temp = root.left;
    return temp;
}
```

هفتاد تک فرزندی نیست پس ازینها هم عبور میکند

```

else {
    Node succParent = root;

    Node succ = root.right;

    while (succ.left != null) {
        succParent = succ;
        succ = succ.left;
    }

    if (succParent != root)
        succParent.left = succ.right;
    else
        succParent.right = succ.right;

    root.key = succ.key;

    return root;
}

```

هفتاد را برابر ساکسور پرنٹ میگیریم ، ۸۰ برابر ساکسور میشود چون از ۷۰ بزرگتر است ، وایل اول اجرا نمیشود ، ساکسور پرنٹ هم مخالف روت نیست پس ایف هم اجرا نمیشد

ساکسور رایت برابر نال است پس ۸۰ نال میشود

و در نهایت ۷۰ که همان روت است برابر میشود با ۸۰

و درخت نهایی ما به صورت زیر است :

```

    |-- 80
      |-- 60
|-- 50
      |-- 40
        |-- 30
          |-- 20

```

هر مثالی بزنیم جواب میگیریم