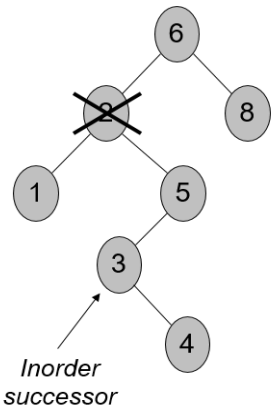
Tree=6, info=2

Treenode * Remove (6, 2)

Treenode *t;

```
t = remove(2, 2);
6->setLeft( t ); //3
```

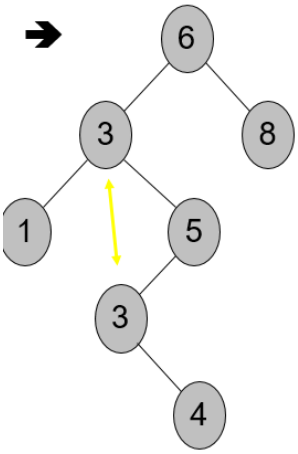
return 6; //return tree



Tree=2, info=2

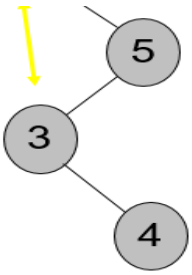
Treenode *t;

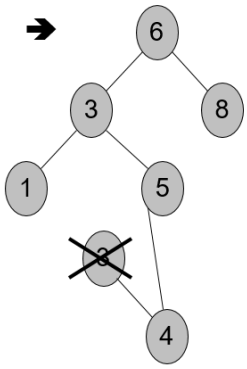
```
TreeNode * minNode;
minNode = findMin(5)      //tree->getRight();
minNode=3//pointer
2->setInfo( 3->getInfo() ); //replace 2 with 3
t = remove(3->getRight(),(minNode->getInfo()));
                Remove(5,3)

3->setRight( t );//5
return tree// 3
```



Tree=5, info=3

```
Treenode *t;
t = remove(3, 3);
 5->setLeft( t ); //5->setLeft(4)
return 5;//tree
```





Treenode *t;

Tree=3, info=3

```
TreeNode* nodeToDelete = 3;
tree=4;//pointer
delete nodetoDelete;
return 4;//tree
```