61050192 ธนพล กฤตินัดดากุล

```java
import java.util.Scanner;
public class BST {
    public static class TNode {
        TNode l = null, r = null, parent = null;
        int data;
        public TNode(int d) {
            data = d;
        }
    }
    static TNode root = null;
    static void insert(int d) {
        TNode q = new TNode(d);
        if (root == null) {
            root = q;
            return;
        }
        TNode p = root;
        while (true) {
            if (d < p.data) { // น้อยกว่าไปซ้าย
                if (p.l == null) {
                    p.l = q;
                    q.parent = p;
                    return;
                } else p = p.l;
            } else { // มากกว่าไปขวา
                if (p.r == null) {
                    p.r = q;
                    q.parent = p;
                    return;
                } else p = p.r;
            }
        }
    }
    static void delete(TNode p) {
        if (p.l == null && p.r == null) { // ไม่มีลูก
            if (p.parent.r == p) {
                p.parent.r = null;
            } else
                p.parent.l = null;
        } else if (p.r == null || p.l == null) { // มีลูกข้างเดียว
            TNode q = (p.l != null) ? p.l : p.r;
            if (p.parent.l == p)
                p.parent.l = q;
            else
                p.parent.r = q;
            q.parent = p.parent;
        } else { // ลูกสอง
            TNode q = p.l;
            while (q.r != null)
                q = q.r;
            delete(q);
            p.data = q.data;
        }
    }
```

```
55      static TNode search(int d) {
56          TNode p = root;
57          while (p != null) {
58              if (p.data == d)
59                  break;
60              if (d < p.data)
61                  p = p.l;
62              else
63                  p = p.r;
64          }
65          return p;
66      }
67      static void inorder(TNode p) {
68          if (p == null)
69              return;
70          inorder(p.l);
71          System.out.println(p.data + " ");
72          inorder(p.r);
73      }
74      public static void main(String[] args) {
75          Scanner sc = new Scanner(System.in);
76          for (int i = 1; i <= 9; i++) {
77              int d = sc.nextInt();
78              insert(d);
79          }
80          inorder(root);
81          delete(root.l);
82          System.out.println("eiei");
83          inorder(root);
84          System.out.println(search(5).l.data);
85      }
86  }
```