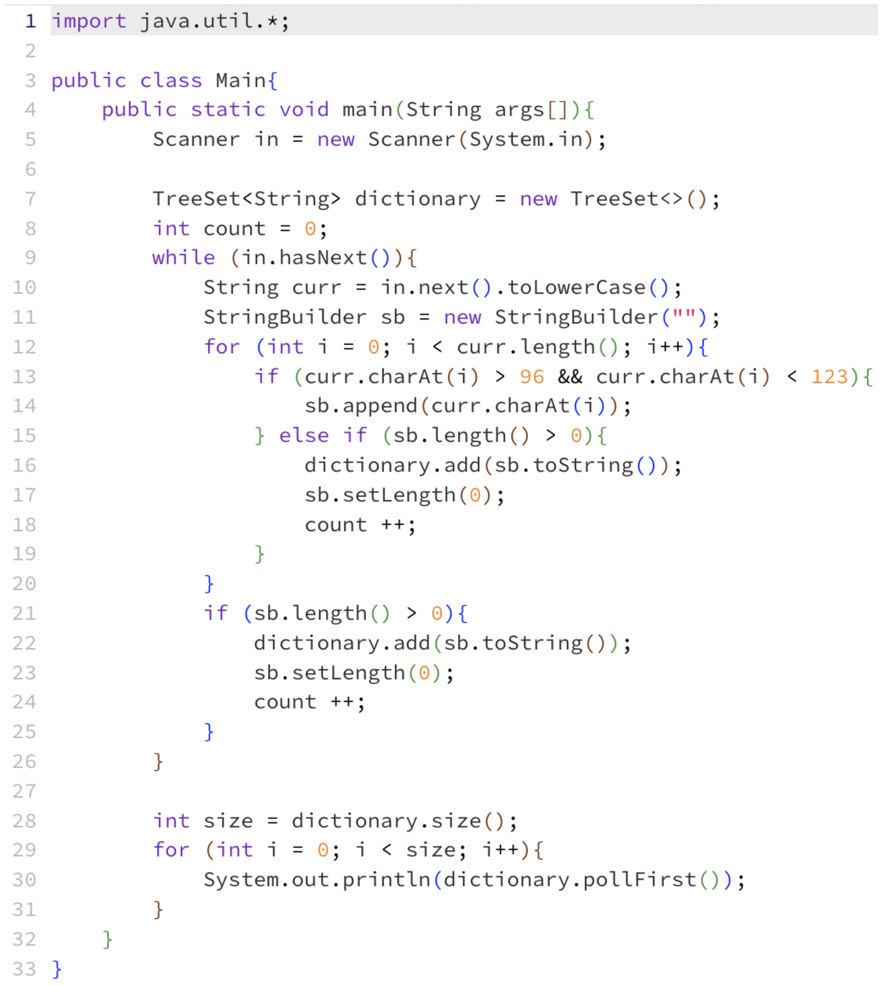
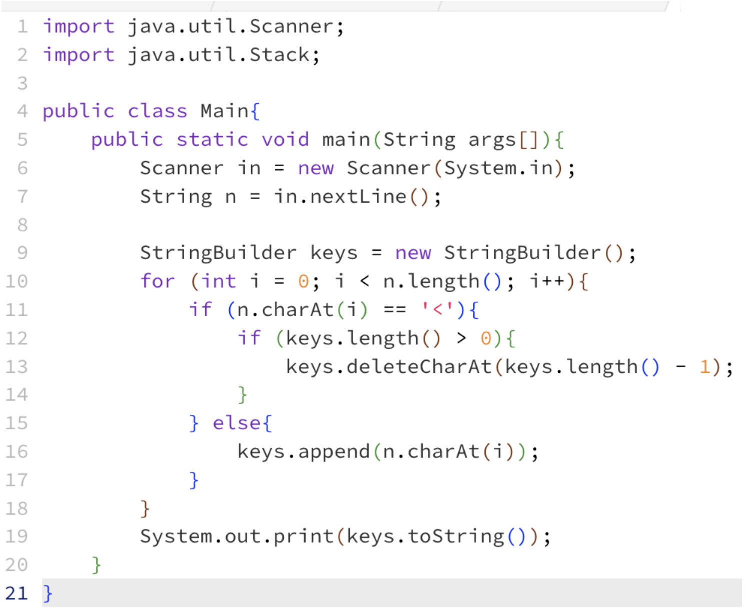
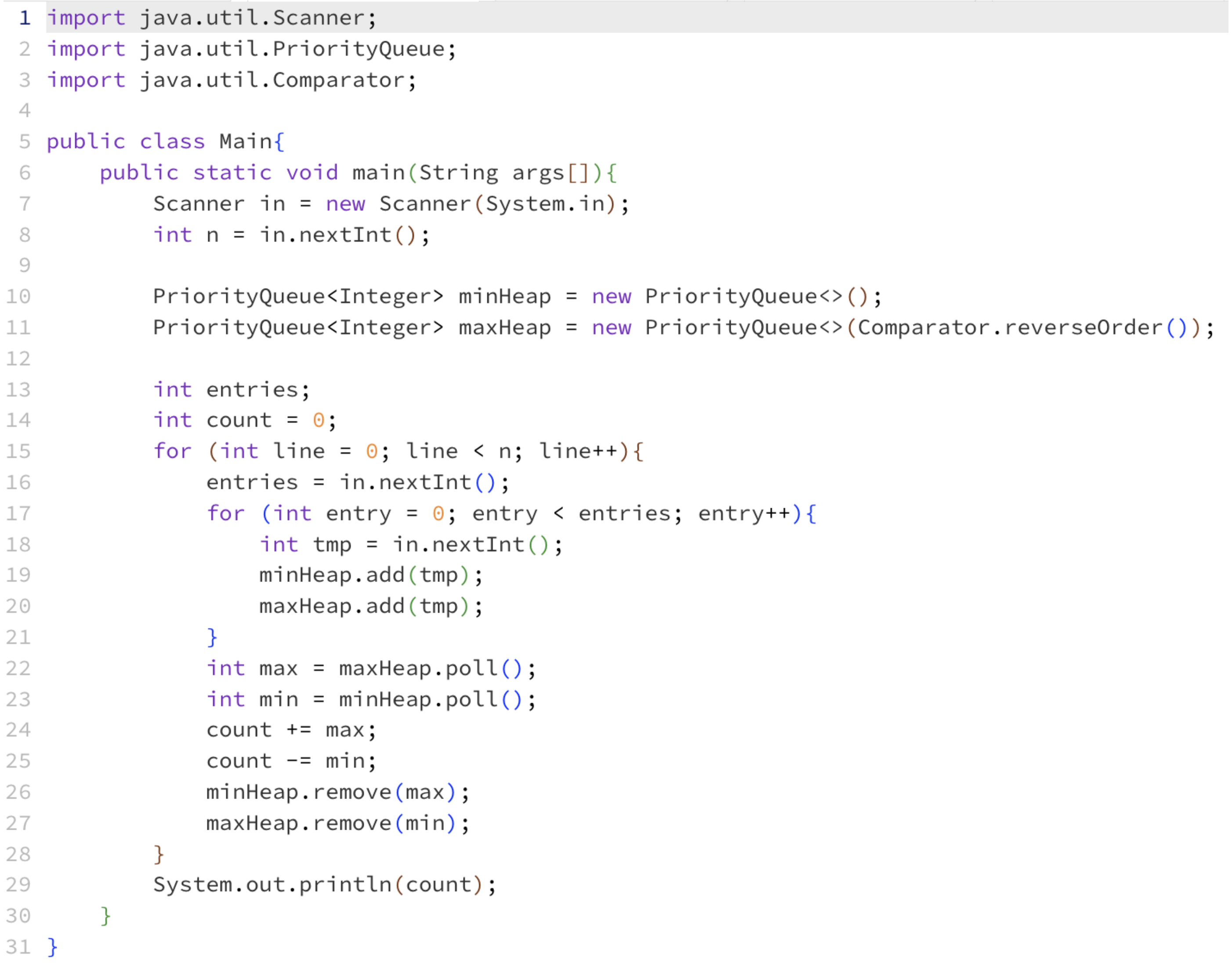


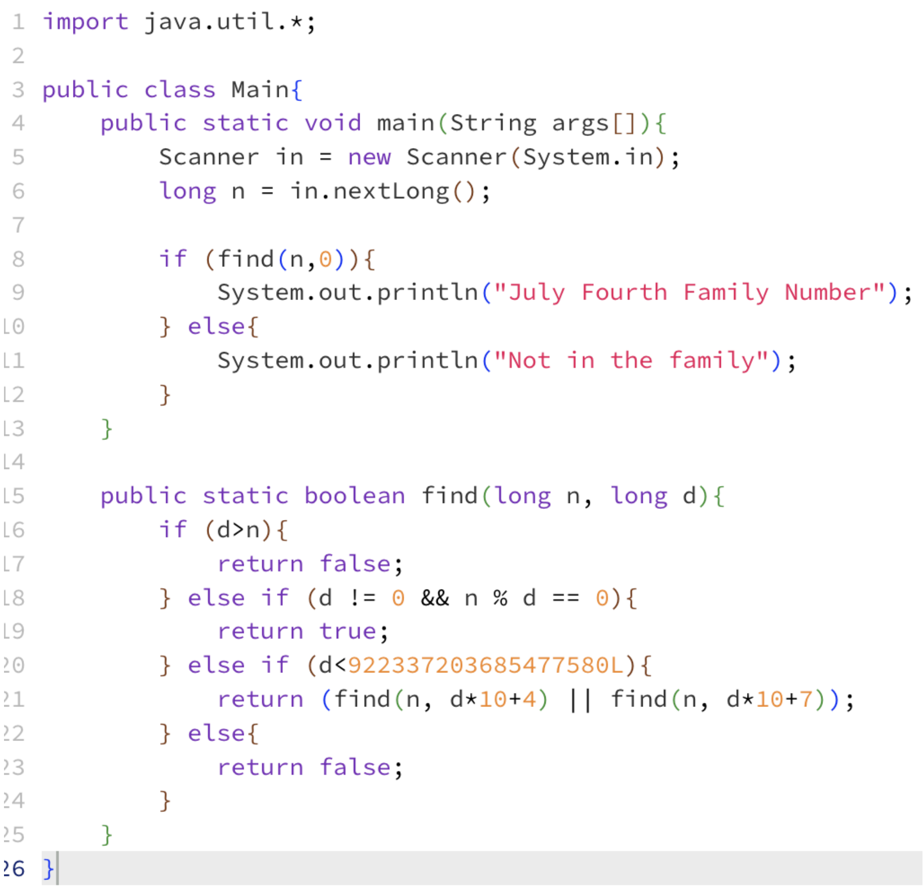
```java
import java.util.*;

public class Main{
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);

        TreeSet<String> dictionary = new TreeSet<>();
        int count = 0;
        while (in.hasNext()){
            String curr = in.next().toLowerCase();
            StringBuilder sb = new StringBuilder("");
            for (int i = 0; i < curr.length(); i++){
                if (curr.charAt(i) > 96 && curr.charAt(i) < 123){
                    sb.append(curr.charAt(i));
                } else if (sb.length() > 0){
                    dictionary.add(sb.toString());
                    sb.setLength(0);
                    count ++;
                }
            }
            if (sb.length() > 0){
                dictionary.add(sb.toString());
                sb.setLength(0);
                count ++;
            }
        }

        int size = dictionary.size();
        for (int i = 0; i < size; i++){
            System.out.println(dictionary.pollFirst());
        }
    }
}
```



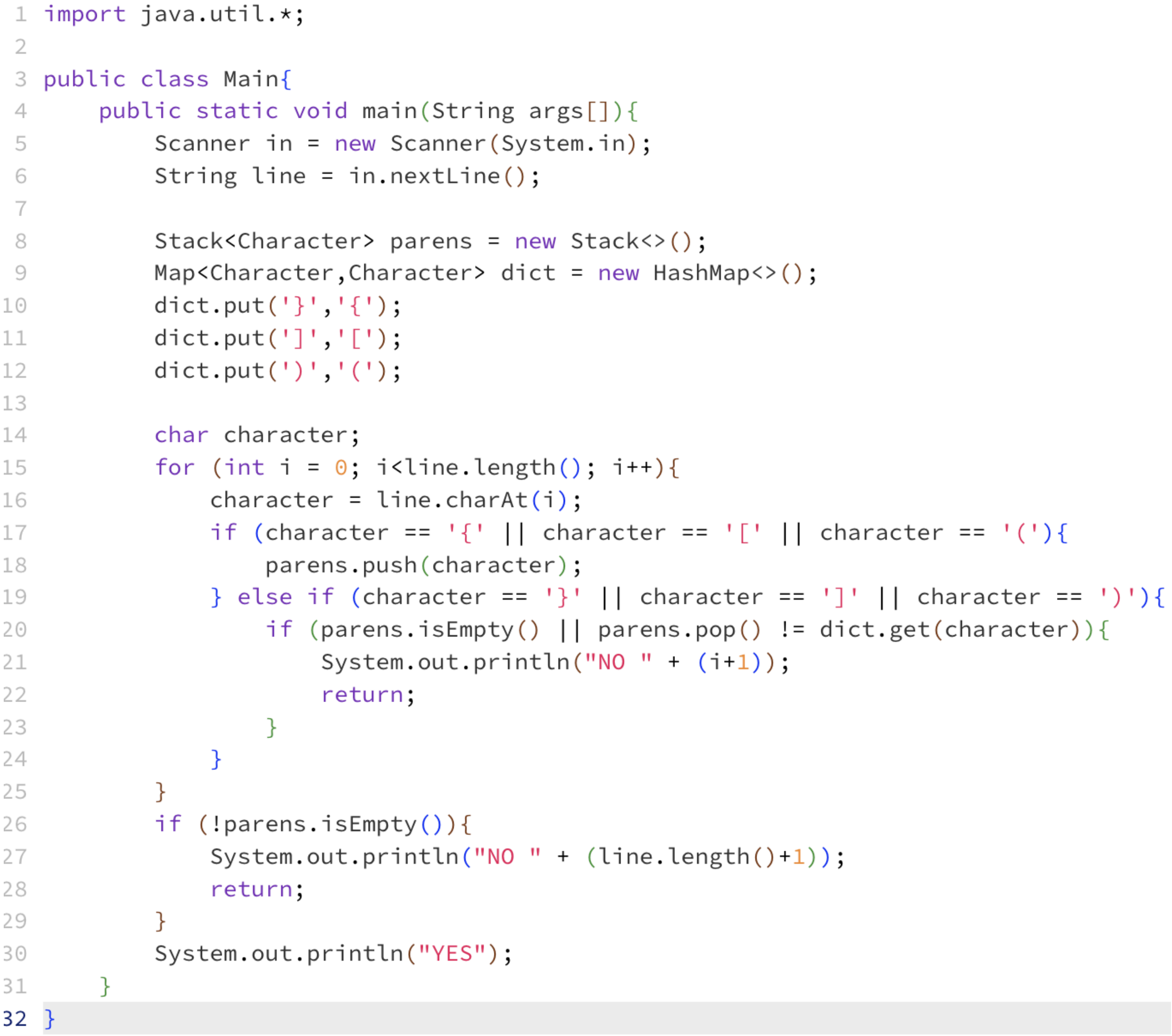
```java
import java.util.Scanner;
import java.util.Stack;

public class Main{
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);
        String n = in.nextLine();

        StringBuilder keys = new StringBuilder();
        for (int i = 0; i < n.length(); i++){
            if (n.charAt(i) == '<'){
                if (keys.length() > 0){
                    keys.deleteCharAt(keys.length() - 1);
                }
            } else{
                keys.append(n.charAt(i));
            }
        }
        System.out.print(keys.toString());
    }
}
```



```java
import java.util.Scanner;
import java.util.PriorityQueue;
import java.util.Comparator;

public class Main{
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);
        int n = in.nextInt();

        PriorityQueue<Integer> minHeap = new PriorityQueue<>();
        PriorityQueue<Integer> maxHeap = new PriorityQueue<>(Comparator.reverseOrder());

        int entries;
        int count = 0;
        for (int line = 0; line < n; line++){
            entries = in.nextInt();
            for (int entry = 0; entry < entries; entry++){
                int tmp = in.nextInt();
                minHeap.add(tmp);
                maxHeap.add(tmp);
            }
            int max = maxHeap.poll();
            int min = minHeap.poll();
            count += max;
            count -= min;
            minHeap.remove(max);
            maxHeap.remove(min);
        }
        System.out.println(count);
    }
}
```



```java
import java.util.*;

public class Main{
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);
        long n = in.nextLong();

        if (find(n,0)){
            System.out.println("July Fourth Family Number");
        } else{
            System.out.println("Not in the family");
        }
    }

    public static boolean find(long n, long d){
        if (d>n){
            return false;
        } else if (d != 0 && n % d == 0){
            return true;
        } else if (d<922337203685477580L){
            return (find(n, d*10+4) || find(n, d*10+7));
        } else{
            return false;
        }
    }
}
```



```java
import java.util.*;

public class Main{
    public static void main(String args[]){
        Scanner in = new Scanner(System.in);
        String line = in.nextLine();

        Stack<Character> parens = new Stack<>();
        Map<Character,Character> dict = new HashMap<>();
        dict.put('}','{');
        dict.put(']','[');
        dict.put(')','(');

        char character;
        for (int i = 0; i<line.length(); i++){
            character = line.charAt(i);
            if (character == '{' || character == '[' || character == '('){
                parens.push(character);
            } else if (character == '}' || character == ']' || character == ')'){
                if (parens.isEmpty() || parens.pop() != dict.get(character)){
                    System.out.println("NO " + (i+1));
                    return;
                }
            }
        }
        if (!parens.isEmpty()){
            System.out.println("NO " + (line.length()+1));
            return;
        }
        System.out.println("YES");
    }
}
```