

题目描述：

给一个正整数 NUM1，计算出新正整数 NUM2，NUM2 为 NUM1 中移除 N 位数字后的结果，需要使得 NUM2 的值最小。

输入描述：

1.输入的第一行为一个字符串，字符串由 0-9 字符组成，记录正整数 NUM1，NUM1 长度小于 32。

2.输入的第二行为需要移除的数字的个数，小于 NUM1 长度。

如：

2615371

4

输出描述：

输出一个数字字符串，记录最小值 NUM2。

如：131

补充说明：

示例 1

输入：

2615371

4

输出：

131

说明：

移除 2、6、5、7 这四个数字，剩下 1、3、1 按原有顺序排列组成 131，为最小值

```
import java.util.*;
```

```
public class Main{
```

```
    public static void main(String[] args){
```

```
        Scanner sc=new Scanner(System.in);
```

```
        String s=sc.nextLine();
```

```
        int n=sc.nextInt();
```

```
        Stack<Integer> st=new Stack<>();
```

```
        for(int i=0;i<s.length();i++){
```

```
            int now =s.charAt(i)-'0';
```

```
            while(!st.empty()&&st.peek()>now&&n>0){
```

```
                st.pop();
```

```
                n--;
```

```
            }
```

```
            st.add(now);
```

```
        }
```

```
        while(!st.empty()&&n>0){
```

```
            st.pop();
```

```
            n--;
```

```
        }
```

```
        String ans="";
```

```
        while(!st.empty()) ans+=st.pop();
```

```
String res=new StringBuffer(ans).reverse().toString();
int i=0;
while(i<res.length()&&res.charAt(i)=='0') i++;
if(res.substring(i).equals("")){
    System.out.println(0);
    return;
}
System.out.println(res.substring(i));
}
}
```