

题目描述：

一贫如洗的樵夫阿里巴巴在去砍柴的路上，无意中发现了强盗集团的藏宝地，藏宝地有编号从  $0 \sim N$  的箱子，每个箱子上面贴有一个数字，箱子排列成一个环，编号最大的箱子的下一个是编号为  $0$  的箱子。

请输出每个箱子贴的数字之后的第一个比它大的数，如果不存在则输出  $-1$ 。

输入描述：

输入一个数字字符串，数字之间使用逗号分隔，例如：  $1,2,3,1$

字符串中数字个数  $\geq 1$ ， $\leq 10000$ ；每个数字值  $\geq -100000$ ， $\leq 100000$

输出描述：

下一个大的数列表，以逗号分隔，例如：  $2,3,6,-1,6$

补充说明：

示例 1

输入：

$2,5,2$

输出：

$5,-1,5$

说明：

第一个  $2$  的下一个更大的数是  $5$ ；

数字  $5$  找不到下一个更大的数；

第二个  $2$  的下一个最大的数需要循环搜索，结果也是  $5$

示例 2

输入:

3, 4, 5, 6, 3

输出:

4, 5, 6, -1, 4

说明:

```
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
```

// 注意类名必须为 Main, 不要有任何 package xxx 信息

```
public class Main {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        // 注意 hasNext 和 hasNextLine 的区别
        String line = in.nextLine();
        String[] split = line.split(",");
        List<Integer> list = new ArrayList<>();
        for(int i=0;i<split.length;i++){
            list.add(Integer.valueOf(split[i]));
        }

        StringBuilder sb = new StringBuilder();

        for(int j=0;j<list.size();j++){
            if(j<list.size()-1){
                sb.append(curMax(list,j)+",");
            }
            else{
                sb.append(curMax(list,j));
            }
        }
        System.out.println(sb.toString());
    }

    private static int curMax(List<Integer> list, int start) {
        int i = start + 1;
        while (true) {
            if (i >= list.size()) {
                i = i % list.size();
            }
            if (list.get(i) > list.get(start)) {
                return list.get(i);
            }
        }
    }
}
```

```
        if (i == start) {  
            return -1;  
        }  
        i++;  
    }  
}
```