

阿里巴巴找黄金宝箱(III)

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。 阿里巴巴念出一个咒语数字，查看宝箱是否存在两个不同箱子，这两个箱子上贴的数字相同，同时这两个箱子的编号之差的绝对值小于等于咒语数字， 如果存在这样的一对宝箱，请返回最先找到的那对宝箱左边箱子的编号， 如果不存在则返回-1。
输入描述：第一行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 ≥ 1 ， ≤ 100000 ；每个数字值 ≥ -100000 ， ≤ 100000 ； 第二行输入咒语数字，例如：3，咒语数字 ≥ 1 ， ≤ 100000 输出描述：存在这样的一对宝箱，请返回最先找到的那对宝箱左边箱子的编号， 如果不存在则返回-1

示例1

输入：6,3,1,6

3

输出：0

说明：

示例2

输入：5,6,7,5,6,7

2

输出：-1

说明：

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

```
// 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
```

```
        while (in.hasNextLine()) { // 注意 while 处理多个 case
```

```
            String a = in.nextLine();
```

```
            String b = in.nextLine();
```

```
            String[] a1 = a.split(" ");
```

```
            String[] b1 = b.split(" ");
```

```
            int[] temp = new int[a1.length/2];
```

```
            int index = 0;
```

```
            HashMap<Integer, Integer> map = new HashMap<>();
```

```
            for (int i = 0; i < a1.length; i+=2) {
```

```
                int j = Integer.parseInt(a1[i]);
```

```
                temp[index]=j;
```

```
                index++;
```

```

        map.put(j,Integer.parseInt(a1[i+1]));
    }
    Arrays.sort(temp);
    ArrayList<List> res = new ArrayList<>();
    for (int i = b1.length-2; i >=0 ; i-=2) {
        ArrayList<Integer> list = new ArrayList<>();
        int pre = Integer.parseInt(b1[i]);
        int next = Integer.parseInt(b1[i+1]);
        for (int j = 0; j < temp.length; j++) {
            int k = temp[j];
            if (k>=pre && k<next){
                Integer orDefault = map.getOrDefault(k, -1111);
                if (orDefault!=-1111){
                    list.add(orDefault);
                    map.remove(k);
                }
            }
        }
        if (list!=null&&list.size()>0){
            res.add(list);
        }else {
            list.add(-1);
            res.add(list);
        }
    }
    if (res!=null&&res.size()>0){
        for (int i = res.size()-1; i >=0; i--) {
            List list = res.get(i);
            StringBuilder sb = new StringBuilder();
            for (Object o : list) {
                sb.append(o).append(" ");
            }
            System.out.println(sb);
        }
    }
}

```