

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

```
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.hasNext()) { // 注意 while 处理多个 case  
            String str = in.next();  
            int number = in.nextInt();  
            String[] split = str.split(",");  
            for (int i = 0; i < split.length; i++) {  
                for (int j = 1; j <= number && i + j < split.length; j++) {  
                    if (Objects.equals(split[i], split[i + j])) {  
                        System.out.println(i);  
                        return;  
                    }  
                }  
            }  
            System.out.println(-1);  
        }  
    }  
}
```