

不开心世纪

时间限制：2.0s 内存限制：256M

题目描述

给定一个长度为 n 的正整数序列 A 。

小 Z 不喜欢序列 A ，他想要从 A 中删除任意多个元素（包括 0 个），将剩余元素保持原有的相对顺序排列，得到一个新序列 B ，使得这个新的序列存在一个长度为 m 的连续子段，并且该子段是一个排列。

小 Z 想知道最少需要在 A 中删除多少元素。

补充定义：

排列：一个序列如果仅包含 $1, 2, \dots, m$ 的数字恰好各一个，那么这是一个长度为 m 的排列。

子段：删除一个序列开头和结尾任意多个元素（包含零）后得到的序列。

输入格式

第一行输入一个整数 T ，表示数据组数。

对于每组数据：

第一行两个正整数 n, m 。

接下来一行 n 个正整数 a_1, a_2, \dots, a_n 表示序列 A 。

令 N_{tot} 表示所有测试组 n 之和。

$1 \leq T \leq 10, N_{tot} \leq 2.1 \times 10^6$ 。

对于每组数据： $1 \leq n, m, a_i \leq 10^6$ 。

输出格式

对于每组数据：

输出一行一个整数表示最少删除元素的数量，如果无论如何都无法出现长度为 m 的子段，则输出 -1 。

样例输入

```
3
6 3
1 1 4 2 2 3
6 3
1 1 4 2 2 7
3 3
3 2 1
```

样例输出

```
2
-1
0
```