

白哥很黑模拟赛

题目名称	小鸡	小 L 过河	黑哥的序列	我比你高 2
题目类型	传统型	传统型	传统型	传统型
目录	chick	river	sequence	taller
可执行文件名	chick	river	sequence	taller
输入文件名	chick.in	river.in	sequence.in	taller.in
输出文件名	chick.out	river.out	sequence.out	taller.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	2.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
测试点数目	10	10	25	25
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	chick.cpp	river.cpp	sequence.cpp	taller.cpp
-----------	-----------	-----------	--------------	------------

编译选项

对于 C++ 语言	-O2 -std=c++14
-----------	----------------

小鸡 (chick)

【题目描述】

你有一个长度为 n 的排列 a 。

你可以对这个排列进行任意多次操作（可以为零次），在每次操作中，你可以选择一个长度为 3 的区间 $[x-1, x+1]$ ，并将这个区间进行一次向右的循环移位。形式化地说，你需要选择一个 $2 \leq x \leq n-1$ ，并将 a_{x-1}, a_x, a_{x+1} 替换为 a_{x+1}, a_{x-1}, a_x 。

现在你想知道，能否利用上述操作将整个排列升序排序。

【输入格式】

从文件 **chick.in** 中读入数据。

本题包含多组测试数据。第一行一个正整数 T ，表示数据组数。

每组数据的第一行，一个正整数 n 表示排列长度。

第二行 n 个正整数，表示排列 a 。

【输出格式】

输出到文件 **chick.out** 中。

对于每组数据，输出一行一个字符串 **Yes** 或 **No**，表示能否将排列升序排序。

【样例 1 输入】

```
1 2
2 4
3 3 2 4 1
4 4
5 2 3 4 1
```

【样例 1 输出】

```
1 Yes
2 No
```

【样例 1 解释】

对于第一组数据，依次操作 $x = [4, 2, 2]$ 即可排序。

对于第二组数据，可以证明不存在合法的方案。

【样例 2】

见选手目录下的 *chick/chick2.in* 与 *chick/chick2.ans*。

【数据范围】

测试点编号	$\sum n \leq$	特殊性质
1, 2	10	无
3, 4	5000	
5, 6, 7, 8, 9		A
10	10^6	无

特殊性质 A：保证对于任意 $1 \leq i \leq n$ ，有 $a_i = n - i + 1$ 。

小 L 过河 (river)

【题目背景】

小 L 要坐船过河。

小 L 不想坐船过河，于是学了魔法。

【题目描述】

小 L 要从位置 1 到达位置 n 。

小 L 有两种魔法，分别是 J 和 S。

若小 L 目前在位置 S ，他施展魔法的效果如下：

- 魔法 J 可以让小 L 的位置加上 x_1 ，即 $S \rightarrow S + x_1$ ，消耗 c_1 点魔力值。
- 魔法 S 可以让小 L 的位置乘上 x_2 ，即 $S \rightarrow S \times x_2$ ，消耗 c_2 点魔力值。

由于小 L 是大魔法家，所以他的魔力是无限的。即便如此，他还是想知道最小的魔力花费是多少。按照惯例，此时你应该帮他解决这个问题。

好过分啊，但是为了得分，你还是得帮帮他。

【输入格式】

从文件 *river.in* 中读入数据。

本题有多组测试数据。输入的第一行两个整数 c, T ，分别表示测试点编号和测试数据组数，接下来输入每组测试数据。样例满足 $c = 0$ 。

对于每组测试数据，输入 5 个整数 n, x_1, c_1, x_2, c_2 ，含义见题目描述。

【输出格式】

输出到文件 *river.out* 中。

对于每组测试数据，单独输出一行一个整数，表示最少消耗的魔力值。若无法到达 n ，输出一行一个 -1。

【样例 1 输入】

```
1 0 4
2 3 2 1 3 1
3 20 7 1 3 1
4 64 10 1 3 1
5 21 2 2 3 7
```

【样例 1 输出】

```
1 1
2 -1
3 -1
4 13
```

【样例 2】

见选手目录下的 *river/river2.in* 与 *river/river2.ans*。

【样例 3】

见选手目录下的 *river/river3.in* 与 *river/river3.ans*。

【样例 4】

见选手目录下的 *river/river4.in* 与 *river/river4.ans*。

【样例 5】

见选手目录下的 *river/river5.in* 与 *river/river5.ans*。

【数据范围】

对于所有测试点，满足：

- $1 \leq T \leq 10^5$;
- $1 \leq n \leq 10^{12}$;
- $1 \leq c_1, c_2 \leq 10^6$;
- $1 \leq x_1, x_2 \leq n$ 。

测试点编号	$T \leq$	$n \leq$
1, 2	10^5	20
3	10	10^4
4, 5		10^{12}
6, 7, 8, 9, 10	10^5	

黑哥的序列 (sequence)

【题目描述】

黑哥很喜欢序列。

黑哥定义一个序列的权值为其中最大值减最小值的差的平方。

现在给定一个长度为 n 的序列 $a_1, a_2, \dots, a_n$ ，问将序列 a 划分成若干个连续子段的权值和的最大值。

【输入格式】

从文件 `sequence.in` 中读入数据。

第一行一个正整数 n ，表示序列 a 的长度。

第二行 n 个正整数，分别表示 $a_1, a_2, \dots, a_n$ 。

【输出格式】

输出到文件 `sequence.out` 中。

一行一个整数，表示划分后权值和的最大值。

【样例 1 输入】

```
1 5
2 5 2 3 4 5
```

【样例 1 输出】

```
1 13
```

【样例 2 输入】

```
1 10
2 537 423 444 130 23 494 57 431 881 436
```

【样例 2 输出】

```
1 1066466
```

【样例 3】

见选手目录下的 *sequence/sequence3.in* 与 *sequence/sequence3.ans*。

【样例 4】

见选手目录下的 *sequence/sequence4.in* 与 *sequence/sequence4.ans*。

【样例 5】

见选手目录下的 *sequence/sequence5.in* 与 *sequence/sequence5.ans*。

【样例 6】

见选手目录下的 *sequence/sequence6.in* 与 *sequence/sequence6.ans*。

【数据范围】

对于所有的测试点，满足： $1 \leq n \leq 10^6, 1 \leq a_i \leq 10^6$ 。

测试点编号	$n \leq$	特殊性质
1	10	无
2 ~ 3	100	
4 ~ 5	1000	
6 ~ 10	5×10^4	
11 ~ 15	10^5	
16		A
17 ~ 20	5×10^5	无
21 ~ 25	10^6	

特殊性质 A: $\forall 1 \leq i < n$, 满足 $a_i \leq a_{i+1}$ 。

我比你高 2 (taller)

【题目背景】

雅礼一男子长得很高。

有诗为证：

《一个人来到雅礼中学》

毫无疑问

我的个子

是全雅礼

最最高的

【题目描述】

雅礼一男子小 X 喜欢数树。

小 X 得到了一棵大小为 n 的树和 k 种颜色，小 L 运用魔法给这个树增加了 m 条边，使其变成了一张图 $G = (V, E)$ ，并向小 X 发起了挑战，要他对满足下列要求的颜色序列 $\{a_n\}$ 计数：

- $\forall 1 \leq i \leq n, 1 \leq a_i \leq k,$
- $\forall (u, v) \in E, a_u \neq a_v.$

换句话说，小 X 需要把这颗树上的每个节点染成 k 种颜色之一，求满足每条边的两个端点颜色不同的方案数。

小 X 虽然是数树高手，但他并不会数图，所以他只能请你帮他求出这个方案数。

由于这个答案可能很大，你只需要告诉小 X 答案对于 $10^9 + 7$ 取模的结果即可。

【输入格式】

从文件 *taller.in* 中读入数据。

第一行四个整数 n, m, k ，分别表示图 G 的节点数、小 L 加的边数和颜色数量。

接下来 $n - 1$ 行，第 i ($1 \leq i \leq m$) 行两个整数 u_i, v_i ，描述一条树上的边，

接下来 m 行，第 i ($1 \leq i \leq m$) 行两个整数 u_i, v_i ，描述一条小 L 加的边。

【输出格式】

输出到文件 *taller.out* 中。

对于每组测试数据，单独输出一行一个整数，表示染色的方案数对 $10^9 + 7$ 取模的结果。

【样例 1 输入】

```
1 5 1 6
2 1 2
3 1 3
4 4 5
5 1 4
6 5 2
```

【样例 1 输出】

```
1 3150
```

【样例 2 输入】

```
1 10 3 15
2 1 8
3 4 1
4 9 6
5 2 9
6 5 4
7 4 10
8 4 7
9 9 5
10 4 3
11 7 10
12 7 5
13 9 3
```

【样例 2 输出】

```
1 499240977
```

【样例 3】

见选手目录下的 *taller/taller3.in* 与 *taller/taller3.ans*。

【样例 4】

见选手目录下的 *taller/taller4.in* 与 *taller/taller4.ans*。

【样例 5】

见选手目录下的 *taller/taller5.in* 与 *taller/taller5.ans*。

【样例 6】

见选手目录下的 *taller/taller6.in* 与 *taller/taller6.ans*。

【样例 7】

见选手目录下的 *taller/taller7.in* 与 *taller/taller7.ans*。

【样例 8】

见选手目录下的 *taller/taller8.in* 与 *taller/taller8.ans*。

【数据范围】

对于所有的测试点，保证：

- $1 \leq n, k \leq 10^5$;
- $0 \leq m \leq 5$ 。

测试点编号	$n \leq$	$m \leq$	$k \leq$
1 ~ 5	2000	0	2000
6 ~ 8	10^5		10^5
9 ~ 10		1	
11 ~ 15	5000	3	
16 ~ 18	10^5	5	
19 ~ 20			