

# 全国青少年信息学奥林匹克竞赛

## NOI2023模拟

时间：7:30-12:20

题目名称	映射	路径	序列
题目类型	传统型	传统型	传统型
目录	xiz	path	seq
可执行文件名	xiz	path	seq
输入文件名	xiz.in	path.in	seq.in
输出文件名	xiz.out	path.out	seq.out
每个测试点时限	2.0秒	1.0秒	2.0秒
内存限制	512 MB	512MB	1024MB
子任务数目	10	10	25
测试点是否等分	是	是	是

### 提交源程序文件名

对于C++语言	xiz.cpp	path.cpp	seq.cpp
---------	---------	----------	---------

### 编译选项

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

### 注意事项与提醒（请选手务必仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C++ 中主函数的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
3. 提交的程序代码文件的放置位置请参照各省的具体要求。
4. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
5. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
6. 程序可使用的栈内存空间限制与题目的内存限制一致。
7. 全国统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz，内存 32GB。上述时限以此配置为准。
8. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以其为准。
9. 终评测时所用的编译命令中不含编译选项之外的任何优化开关。

# 映射 (xiz)

## 【问题描述】

给定字符串  $A$  和  $B$ 。

字符串  $A$  和字符串  $B$  匹配的定义为：存在一个字符的映射，使得  $A$  应用这个映射之后等于  $B$ ，且这个映射必须为一个排列。

$A = 121, B = 313$ ，当映射为  $\{1 \rightarrow 3, 2 \rightarrow 1, 3 \rightarrow 2\}$  时  $A' = B$ ，可以匹配

$A = 212, B = 313$ ，当映射为  $\{1 \rightarrow 1, 2 \rightarrow 3, 3 \rightarrow 2\}$  时  $A' = B$ ，可以匹配

$A = 232, B = 313$ ，当映射为  $\{1 \rightarrow 2, 2 \rightarrow 3, 3 \rightarrow 1\}$  时  $A' = B$ ，可以匹配

$A = 123, B = 111$ ，当映射为  $\{1 \rightarrow 1, 2 \rightarrow 1, 3 \rightarrow 1\}$  时  $A' = B$ ，但此时映射不为一个排列，不能匹配。

求  $A$  的哪些连续子串与  $B$  匹配。

## 【输入格式】

第一行两个整数  $T, C$ ，分别表示数据组数与字符集大小；

对于每组数据，第一行两个整数  $n, m$ ，分别表示  $A, B$  的长度；

第二行  $n$  个整数，第  $i$  个整数表示  $A_i$ ；

第三行  $m$  个整数，第  $i$  个整数表示  $B_i$ ；

## 【输出格式】

对于每组数据输出两行，第一行一个整数  $k$ ，表示匹配的个数；

第二行  $k$  个整数，表示匹配的子串在  $A$  中的开始位置 (下标从 1 开始)，升序排列。

## 【样例输入1】

```
3 3
6 3
1 2 1 2 3 2
3 1 3
6 3
1 2 1 2 1 2
3 1 3
6 3
1 1 2 1 2 1
3 1 3
```

## 【样例输出1】

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

## 【样例2】

见选手目录下的 *ex\_xiz2.in* 与 *ex\_xiz2.ans*。

## 【数据范围及约定】

对于前 10% 的数据,  $n, m, C \leq 1000, T \leq 3$ ;

对于前 30% 的数据,  $n, m \leq 100000, C \leq 40, T \leq 3$ ;

对于前 60% 的数据,  $n, m, C \leq 100000, T \leq 3$ ;

对于 100% 的数据,  $n, m, C \leq 1000000, T \leq 3$ ;

# 路径 (path)

## 【问题描述】

给出一个包含  $n + 1$  个结点的有向图，结点的编号为  $0$  到  $n$ 。图中有  $m$  条有向边，第  $i$  条有向边起点为  $u_i$ ，终点为  $v_i$ ，且长度为  $w_i$ 。并且这些边还满足如下的性质：

- 对任意一条边，满足  $u_i < v_i$ 。
- 不存在两条边  $i, j$  使得  $u_i < u_j < v_i < v_j$ 。

除了结点  $0$  和结点  $n$  以外，其余的每个结点都有颜色。现在需要你找出一条从结点  $0$  走到结点  $n$  的最短路径。对于任意一种颜色，这条路径要么经过了这种颜色的所有结点，要么就不经过这种颜色的任意一个结点。如果不存在这样的路径，请输出  $-1$ ，否则输出最短路径的长度。

## 【输入格式】

输入第一行为两个整数  $n, m$ 。

第二行为  $n - 1$  个整数  $c_1, c_2, \dots, c_{n-1}$ ，依次表示每个结点的颜色。不同的整数表示不同的颜色。

接下来  $m$  行，每行三个整数  $u_i, v_i, w_i$ 。

## 【输出格式】

输出一个整数表示最短路径的长度。如果不存在这样的路径则输出  $-1$ 。

## 【样例输入1】

```
3 3
1 1
0 1 10
0 2 10
2 3 10
```

## 【样例输出1】

```
-1
```

## 【样例输入2】

```
7 11
1 2 3 3 1 3
0 1 1
1 2 1
2 4 1
4 5 1
5 6 1
6 7 1
0 2 10
2 5 2
5 7 3
2 7 1
3 4 1
```

### 【样例输出2】

```
7
```

### 【样例3】

见选手目录下的 *ex\_path2.in* 与 *ex\_path2.ans*。

### 【数据范围及约定】

对于 20% 的数据，保证  $1 \leq n \leq 10$ ， $1 \leq m \leq 15$ 。

对于 40% 的数据，保证  $1 \leq c_i \leq 10$ 。

对于 100% 的数据，保证  $1 \leq n, m, c_i \leq 1000$ ， $0 \leq u_i \leq v_i \leq n$ ， $1 \leq w_i \leq 10^6$ 。

# 序列 (seq)

## 【问题描述】

给定一个长度为  $n$  的正整数序列  $a_1, \dots, a_n$ , 现在有  $m$  次操作, 分为两种:

- $1\ l\ r\ t$ : 将区间  $[l, r]$  降序排序 ( $t = 0$ ) 或升序排序 ( $t = 1$ )
- $2\ l\ r$  询问区间  $[l, r]$  内元素之积的十进制下最高位

## 【输入格式】

第一行两个整数  $n, m$ ;

接下来  $m$  行, 每行四个或三个整数, 表示一个操作;

## 【输出格式】

对于每个询问操作输出一行一个整数, 表示答案。

## 【样例输入1】

```
10 11
1 5 10 3 9 2 8 6 4 7
2 5 8
1 1 3 0
2 2 6
1 6 8 1
2 1 10
2 4 9
1 2 5 0
1 4 8 1
1 3 6 0
2 1 10
2 3 8
```

## 【样例输出1】

```
8
2
3
1
3
1
```

## 【样例2】

见选手目录下的 *ex\_seq2.in* 与 *ex\_seq2.ans*。

## 【数据范围及约定】

对于前 20% 的数据, 不存在 1 操作;

对于另 20% 的数据,  $n, m \leq 1000$ , 保证询问的区间的元素积  $< 2^{63}$ ;

对于另 20% 的数据,  $n, m \leq 1000$ ;

对于 100% 的数据,  $1 \leq n, m \leq 200000$ ,  $1 \leq a_i \leq n$ ;

**由于可能存在的精度问题, 对于后 40% 的数据, 只要与标准答案不同的地方不超过 20 个即视为正确。**