

# 全国青少年信息学奥林匹克竞赛

## NOIP2023模拟

时间：7:50-12:20

题目名称	序列	树上排列计数	无向图匹配计数	机器人
题目类型	传统型	传统型	传统型	传统型
目录	sequence	count	graph	robot
可执行文件名	sequence	count	graph	robot
输入文件名	sequence.in	count.in	graph.in	robot.in
输出文件名	sequence.out	count.out	graph.out	robot.out
每个测试点时限	1.0秒	1.0秒	2.0 秒	1.0秒
内存限制	512 MB	512 MB	512MB	512 MB
子任务数目	Subtask	Subtask	Subtask	20
测试点是否等分	否	否	否	是

### 提交源程序文件名

对于C++语言	sequence.cpp	count.cpp	graph.cpp	robot.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）

- 文件名（程序名和输入输出文件名）必须使用英文小写。
- 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。
- 提交的程序代码文件的放置位置请参照各省的具体要求。
- 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。
- 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）。
- 程序可使用的栈内存空间限制与题目的内存限制一致。
- 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。
- 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。
- 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。

# 序列 (sequence)

## 【问题描述】

有一个长度为  $n$  的 01 序列  $a$ ，其中部分位置已给定。你可以在每个未给定的位置中任意选择填入 0 还是 1。

定义一个填充方案是好的当且仅当恰好存在  $m$  个不同的  $i$  满足  $1 \leq i < n$  且  $a_i \neq a_{i+1}$ 。

对于长度为  $n$  的序列  $p, q$ ，定义  $p$  的字典序严格小于  $q$  的字典序当且仅当存在一个  $1 \leq i \leq n$  满足  $a_i < b_i$  且所有  $1 \leq j < i$  都满足  $a_j = b_j$ 。

你需要求出所有好的填充方案中字典序最小的一个。如果无解则输出  $-1$ 。

## 【输入格式】

第一行，共两个整数，表示  $n, m$ 。

第二行，共一个长度为  $n$  的只包含 0, 1, ? 的字符串。第  $i$  位为 0 表示给定  $a_i = 0$ ，第  $i$  位为 1 表示给定  $a_i = 1$ ，第  $i$  位为 ? 表示  $a_i$  未给定。

## 【输出格式】

共一行，一个长度为  $n$  的 01 序列，表示答案。相邻两个元素之间不加空格。

## 【样例输入1】

```
5 3
0??1?
```

## 【样例输出1】

```
01011
```

## 【样例2】

见选手目录下的 `sequence/sequence2.in` 与 `sequence/sequence2.ans`。

## 【数据范围及约定】

对于 100% 的数据， $1 \leq m < n \leq 10^6$ 。

Subtask1(10%):  $n \leq 20$ 。

Subtask2(20%):  $n \leq 5 \times 10^3$ 。

Subtask3(20%): 所有  $a_i$  均未给定。

Subtask4(50%): 无特殊限制。

## 2. 树上排列计数 (count)

### 【题目描述】

给定一个  $n$  个点的有根树  $T$ ，根为 1。

对于一个  $1 \sim n$  的排列  $a$ ，设  $k$  表示有多少个点  $u$  满足它的所有祖先  $v$  都有  $a_u \leq a_v$ 。我们定义这个排列的权值为  $m^k$ 。

你需要求出所有排列的权值之和，答案对 998244353 取模。

### 【输入格式】

第一行，共两个整数，表示  $n, m$ 。

第二行，共  $n - 1$  个整数，第  $i$  个数表示点  $i + 1$  的父亲节点。

### 【输出格式】

输出共一行，一个整数，表示答案。

### 【样例输入1】

```
4 2
1 1 1
```

### 【样例输出1】

```
180
```

### 【样例2】

见选手目录下的 *count/count2.in* 与 *count/count2.ans*。

### 【数据范围及约定】

对于 100% 的数据， $1 \leq n \leq 5 \times 10^3, 1 \leq m < 998244353$ 。

Subtask1(10%):  $n \leq 10$ 。

Subtask2(20%): 保证  $T$  为一条链。

Subtask3(30%):  $n \leq 50$ 。

Subtask4(10%):  $n \leq 300$ 。

Subtask5(30%): 无特殊限制。

### 3.无向图匹配计数 (graph)

#### 【题目描述】

给定一个  $n$  个点  $m$  条边的简单无向图  $G = (V, E)$ 。

$G$  满足如下限制:  $\forall (u, v) \in E$ , 有  $|u - v| = c_1$  或  $|u - v| = c_2$ 。

在  $G$  中选择若干条边 (可以选 0 条), 要求选择的边中任意两条均无公共点。

你需要求出有多少种不同的选择方案。答案对 998244353 取模。

#### 【输入格式】

第一行, 共四个整数, 表示  $n, m, c_1, c_2$ 。

接下来  $m$  行, 每行两个整数  $u, v$ , 表示  $G$  中的一条边  $(u, v)$ 。

#### 【输出格式】

输出共一行, 一个整数, 表示答案。

#### 【样例输入1】

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

#### 【样例输出1】

```
5
```

#### 【样例2】

见选手目录下的 *graph/graph2.in* 与 *graph/graph2.ans*。

#### 【数据范围及约定】

对于 100% 的数据,  $1 \leq n \leq 200, 1 \leq m \leq 2n, 1 \leq c_1 < c_2 < n, 1 \leq u < v \leq n$ 。

Subtask1(10%):  $c_1, c_2 \leq 15$ 。

Subtask2(20%):  $n \leq 60$ 。

Subtask3(20%):  $n \leq 100$ 。

Subtask4(20%):  $n \leq 140$ 。

Subtask5(30%): 无特殊限制。

## 4. 机器人 (robot)

### 【题目描述】

你的人类铲屎官得到了一个机器人。这个机器人有点迟钝,  $T$  个时刻没有接到指令就会睡着。人类想要通过实验得到  $T$  的值。

机器人在一条长度为  $L$  的轨道上运动, 轨道上有  $0 \sim L$  的坐标。人类可以给机器人发出向前一步 (若坐标  $< L$  则坐标  $+1$ , 否则原地不动) 或者向后一步 (若坐标  $> 0$  则坐标  $-1$ , 否则原地不动) 的指令。之前提到这个机器人有点迟钝, 具体地讲, 如果最后一次接到指令是在时刻  $t$ , 那么, 从时刻  $t + T + 1$  开始, 它就会处于睡眠状态。处于睡眠状态的机器人, 在接到一条指令后, 不会执行它, 而只会变成清醒 (.....) 的状态。

人类做了一个实验, 实验中, 他向机器人发出的每个指令都以  $+t$  或者  $-t$  的形式给出, 代表人类在时刻  $t$  向机器人发出了前进一步/后退一步的指令。所有指令结束后, 机器人停在了位置  $P$ 。

你的这位人类比较废柴, 忘记了机器人在  $0$  时刻的坐标  $O$ , 感到束手无策。于是他求助于你, 想知道  $T$  最大可能是多少, 如果  $T$  可以是无限大, 就输出 *infinity*。

此外, 他还希望知道在这个  $T$  下, 坐标  $O$  的最大可能值是多少。

机器人一开始处于休眠状态。

### 【输入格式】

第一行包含三个整数  $n, L, P$ ,  $n$  表示指令的个数,  $L$  表示轨道的长度,  $P$  表示最终机器人停留的坐标。

接下来  $n$  行, 每行有一个字符  $c$  和一个数字  $t$ , 中间用一个空格隔开。  $c \in \{'+', '-'\}$ , 表示在时刻  $t$ , 人类给出了前进一步或后退一步的指令。

### 【输出格式】

若  $T$  可以是无限大, 输出一行一个单词 *infinity*。

否则输出一行两个数  $T, O$ , 分别表示休眠间隔  $T$  的最大可能值, 以及在这个  $T$  下,  $O$  的最大可能值, 两个数中间用一个空格隔开。

### 【样例输入1】

```
6 5 4
+ 1
+ 5
- 9
+ 10
- 11
- 13
```

### 【样例输出1】

```
1 5
```

### 【样例1解释】

初始机器人位于 5

时刻 1 时接受指令，变为工作状态。时刻 3 起再次休眠。

时刻 5 时接受指令，变为工作状态。时刻 7 起再次休眠。

时刻 9 时接受指令，变为工作状态。

时刻 10 时接受向前一步的指令，由于已经位于  $L = 5$  处，原地不动。

时刻 11 时接受向后一步的指令，到达坐标 4 处，时刻 13 起再次休眠。

时刻 13 时接受指令，变为工作状态。时刻 15 起再次休眠。

最终停留在坐标 4 处。

### 【样例2】

见选手目录下的 *robot/robot2.in* 与 *robot/robot2.ans*。

### 【数据范围及约定】

对于前 10% 的数据， $L, n \leq 100, t \leq 4000$ ；

对于前 20% 的数据， $L, n \leq 100, t \leq 10^9$ ；

对于前 50% 的数据， $L \leq 100, n \leq 10000, t \leq 10^9$ ；

对于前 100% 的数据， $2 \leq n \leq 100000, 2 \leq L \leq 5000, 0 \leq p \leq L, 0 \leq t \leq 10^9$ ，且输入的  $t$  严格递增。