

模拟赛

题目名称	联合权值	乘积最大	点	马
题目类型	传统题	传统题	传统题	传统题
可执行文件名	link	cjzd	point	horses
输入文件名	link.in	cjzd.in	point.in	horses.in
输出文件名	link.out	cjzd.out	point.out	horses.out
每个测试点时限	1s	1s	1s	1s
运行内存上限	512MB	512MB	512MB	512MB

编译命令：

C++	-Wl,--stack=1073741824 -O2 -std=c++14
其它语言	rm -r -f

注意事项：

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++ 中函数 main() 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. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
4. 选手应将各题的源程序放在选手文件夹内，**不要建立子文件夹**。

联合权值(link)

题目描述

有些时候爱与爱人之间会有一定的联系...

有些人也会爱着另外一个人...

有些人的爱可以被衡量但是是无价的...

legendgod 也会爱人, 对于事件 i 爱的深沉与否可以由 w_i 来评价。

爱是会传播的, 爱也是简单的是**没有环**的。

在 legendgod 短暂的人生中, 有 n 个事件, 有 $n - 1$ 条纽带连接着不同的事件。对于事件 (u, v) 虽然说可能有大大小小的意外, 但是总归可以看成一条纽带, 也就是说任何事件之间的纽带**最多只有一条**。

爱是不一定是直接相连接的, 但是爱隔着太远也就成了奢望。

真正的爱, 我们称其为 **爱意**, 其可以表示为两个事件 (u, v) 其距离恰好为 2。

当然 **爱意** (u, v) 和 (v, u) 是没有区别的。

legendgod 想知道他一生中有多少 **爱意**。

输入格式

第一行包含 1 个整数 n 。

接下来 $n - 1$ 行, 每行包含 2 个用空格隔开的正整数 u, v , 表示编号为 u 和编号为 v 的事件之间有边相连。

最后 1 行, 包含 n 个正整数, 每两个正整数之间用一个空格隔开, 其中第 i 个整数表示图 G 上事件 i 的爱的深沉评估为 w_i 。

输出格式

输出共 1 行, 包含 2 个整数, 之间用一个空格隔开, 依次为图 G 上爱意的最大值和所有爱意权值之和, 对 10007 取余。

样例数据

input1

```
5
1 2
2 3
3 4
4 5
1 5 2 3 10
```

output1

```
20 74
```

数据规模与约定

对于 30% 的数据, $1 < n \leq 100$

对于 60% 的数据, $1 < n \leq 2000$

对于 100% 的数据, $1 < n \leq 200000, 0 < w_i \leq 10000$ 。

保证一定存在可产生爱意的有序点对。

乘积最大 (cjzd)

题目描述

重聚是一个人的狂欢，分离是一群人的舞蹈...

有些事情总是会分离的，legendgod 和其夫人也是如此...

从其认识到谈恋爱，总共有 n 天。但是却有 K 次被迫分离。

分离是一群人的舞蹈，两人每天的窃窃私语次数可以看做 $x_i, 0 \leq x_i \leq 9$ 。

不妨称这个串为 言串。

总共有 K 次被迫分离，每次分离本质上就是将之前 言串 的某个位置插入 \times 。

称 言语值 为插入恰好 K 个 \times 的时候 \times 左右数的乘积的最大值。

举个例子：有一个 言串：312，当 $N = 3, K = 1$ 时会有以下两种分法：

1. $3 \times 12 = 36$

2. $31 \times 2 = 62$

这时，符合题目要求的结果是： $31 \times 2 = 62$ 。

输入输出格式

输入格式

程序的输入共有两行：

第一行共有 2 个自然数 N, K ($6 \leq N \leq 40, 1 \leq K \leq 6$)

第二行是一个长度为 N 的数字串。

输出格式

结果显示在屏幕上，相对于输入，应输出所求得的 言语值（一个自然数）。

输入输出样例

输入样例 #1

```
4 2
1231
```

输出样例 #1

62

点(point)

题目描述

相逢可能并非是问候...

初恋的事情是谁都不能决定的...

legendgod 从遇见到热爱，是一种对人，对事的区分与了解。

有 n 个事件，互不相关。legendgod 经过了 m 次意外，或者说是相逢。

1. 让事件 i, j 产生联系。
2. legendgod 遇到了悲伤的事情，比如分手了。决定断绝最后产生联系的 k 个事件对。

我们称厌恶值为互相没有联系事件的集合大小的最大值，厌恶个数为满足刚才条件的方事件集合个数。

对 998244353 取模。

输入格式

第一行两个整数 n, m 。接下来 m 行每行第一个数表示事件类型，接下来 2 或 1 个数表示 i, j 或 k 。

输出格式

对于每个操作，输出一行两个整数，用一个空格隔开。

样例数据

input

```
3 7
1 1 2
1 1 3
1 3 3
2 1
1 1 2
2 2
2 1
```

output

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

数据规模与约定

对于 100% 的数据, $n, m \leq 500000$

马 (horses)

题目描述

每个人不论男女都是自己的王子或者公主。

legendgod 可能也是如此吧。

王子和公主会结婚，会一起在公堂上处理事务，会在皇宫的亭子边上乘凉，但是事务总是不能丢却的，这也好，分工罢了。

每年的事务是有限的，老话说得好 事会生事，如果第 i 年有 h 件事务，在那年末会有 $h \times x_i$ 件事务等待处理，当然事情不会无缘无故消失，所以 $x_i \geq 1$ 。

legendgod 即便是城邦之中最智慧的人，但是却往往忙于了解民声。而这重任只能放到其夫人 fym 身上了。

fym 是顶智慧的，但是事情会发酵，每年处理事情都会增加民众的好感度，如果第 i 年处理了一件事情那么好感度会增加 y_i ，显然处理完之后事情就消失了，毕竟是为民服务，每年的好感度至少为 1。

fym 毕竟只是兼职，真正的职位可是公主呢！所以她只能在当年末处理事情，也就是在该年的事情增加之后处理。

假设其在位时间为 n 年。

legendgod 人如其名，在位也是不是随随便便就可以结束的。

好的君王总是想要民众生活富足，希望作为大臣的你来算一下怎么样处理事情才可以让好感值最大。

处理事情都是假设，legendgod 和 fym 都还是王子公主，还在热恋期间怎么可能成为真正的君王呢？这一切都是成长路上的考验，所以假想的事情变化 x_i 和好感度 y_i 会有 m 次变化，每次变化改变一个 x_i 或 y_i ，你需要对于每个修改求出最大好感度，当然这个太大了，所以输出的时候 $\text{mod } 10^9 + 7$ 就行了。

输入输出格式

输入格式

- 第 1 行，一个整数 N ，表示总共有 N 年。
- 第 2 行，共 N 个正整数 $X[0], \dots, X[N - 1]$ ，对于 $0 \leq i \leq N - 1$ ， $X[i]$ 表示 i 年的事情变化系数。

- 第 3 行, 共 N 个正整数 $Y[0], \dots, Y[N - 1]$, 对于 $0 \leq i \leq N - 1$, $Y[i]$ 表示 i 年末处理一件事情的好感度。
- 第 4 行, 一个整数 M , 表示更新次数。
- 第 5, \dots , $M + 4$ 行, 每行 3 个数字 $type, pos, val$ ($type = 1$ 表示更改 $X[pos]$ 为 val , $type = 2$ 表示更改 $Y[pos]$ 为 val) 。

输出格式

- 共 $M + 1$ 行
- 第 1 行: 一个整数表示初始状态下, 最大好感度模 $10^9 + 7$ 后的值。
- 第 2, \dots , $M + 1$ 行: 每行一个整数, 表示这次更新后最大好感度模 $10^9 + 7$ 后的值。

输入输出样例

输入样例 #1

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

输出样例 #1

```
8
6
```

说明

对于 100% 的数据, $1 \leq N \leq 5 \times 10^5$, $0 \leq M \leq 10^5$, $x_i, y_i < 10^9 + 7$

有些东西只会藏在心里。

爱是会不断继续的, 有些时候即使有了时间空间的间隔, 但是这种纽带是永存的。