

# trick or treat

2018 年 10 月 31 日 08:30 - 12:00

## 题目概况

题目名称	冬幕节序曲	小U的女装	萨温节的信封
源程序文件名	leset.c/cpp/pas	femaledress.c/cpp/pas	envelope.c/cpp/pas
每个测试点时限	1 秒	1 秒	1 秒
测试点数目	10	25	50
每个测试点分值	10	4	2
运行内存上限	512MiB	512MiB	512MiB
结果比较方式	全文比较	Special Judge	全文比较

结果比较方式为全文比较的题目会过滤行末回车和空格。

## 编译命令（以第一题为例）

- 对于 C 语言: `gcc -o leset leset.c -O2 -lm`
- 对于 C++ 语言: `g++ -o leset leset.cpp -O2 -lm`
- 对于 Pascal 语言: `fpc leset.pas -O2`

# Problem A. 冬幕节序曲

## 题目背景

众所周知, Leaves 是个善良可爱爱党爱国友善敬业严谨严肃认真友善体贴暖心感性而又不失理性的人。

## 题目描述

一开始, Leaves 手上有含有三个元素的可重集合:

$$\left\{\frac{0}{1}, \frac{1}{1}, \frac{1}{0}\right\}$$

其中  $\{\frac{1}{0}\}$  可看作是无穷大。

在很久很久以前, Leaves 最喜欢的数是  $\frac{1}{1}$ 。众所周知, 男人都是善变的。你需要支持以下两个操作。

- **L**操作, 选中这个集合中, 小于等于当前你喜欢的数的最大的数。
- **R**操作, 选中这个集合中, 大于等于当前你喜欢的数的最小的数。

若你当前最喜欢的数是  $\frac{a}{b}$ , 选中的数字是  $\frac{c}{d}$ , 则将  $\frac{a+c}{b+d}$  约分后放入这个集合。众所周知, 男人都是喜新厌旧的。所以会把新加入的这个数作为自己喜欢的数。

但是 Leaves 不是这样子的人, 所以 Leaves 就把这个任务交给看起来更合适的你啦, 他想请你回答, 最喜欢的那个数字时多少?

## 输入格式

第一行一个整数。表示字符串  $S$  长度。第二行一个字符串  $S$ , 包含若干个复合操作。一个复合操作由两部分组成。操作串和重复次数。具体表现形式为一个由 **L** 和 **R** 组成的字符串和一个整数。例如 **LRL3**, 其意义为将 **LRL** 这个操作重复三次。

## 输出格式

一行一个整数。

表示答案对 998244353 取模的结果: 即设答案化为最简分数后的形式为  $\frac{a}{b}$ , 其中  $a$  和  $b$  互质, 你需要输出整数  $x$  使得  $bx \equiv a \pmod{998244353}$  且  $0 \leq x \leq 998244352$ 。(可以证明这样的  $x$  是唯一的)

## 样例

### 样例输入1

```
1 | 2
2 | R4
```

### 样例输出1

```
1 | 5
```

### 样例输入2

```
1 | 6
2 | L1R2L1
```

### 样例输出2

```
1 | 285212673
```

## 限制

对于 10% 的数据，保证字符串中只会出现 **R** 或只出现 **L**。

对于另外 40% 的数据，保证  $|S| \leq 10^5$ ，所有出现数字  $\leq 10$ 。

对于 100% 的数据，保证  $|S| \leq 10^6$ ，所有出现数字  $\leq 10^9$ 。

## Problem B. 小U的女装

### 题目背景

近年来，S 国的小U立过了无数的女装 flag。他为了偿还，决定去 S 国的首都 Girlfriend 城卖女装。

### 题目描述

GirlFriend 城由  $n$  个小镇组成，其中小镇之间有  $m$  条双向道路，每条道路连接着两个小镇。不过由于 Girlfriend 城的城市建设者因为找不到 Girlfriend 而无心工作，**不保证任意两个小镇可以通过这些双向道路直接或间接地到达**。其中每一个小镇和每一条道路的中间都设有一个女装店。在每家店最多只能购买一件女装。而道路中间的女装店的店主与道路两端的女装店的店主构成竞争关系，如果他在道路中间和道路两端中的一端同时买了女装，那么这两家店就会由于嫉妒抛弃自己的Girlfriend打起来。因此，**如果小 U 在道路上的女装店买女装，就不能在道路连接的两个小镇买女装**。

现在小 U 想知道他自己最多能够买多少件女装，以及他有多少种方案能够买到这么多女装。可是小 U 已经有一段时间不去处理这些有趣的问题了，所以，他想让你帮他解决这两个问题。由于第二个问题的答案太大，请你输出答案模 998244353 的结果。

### 输入格式

第一行两个用空格分隔的正整数  $n, m$ ，分别表示 Girlfriend 城的小镇数和道路数。

接下来  $m$  行，第  $i$  行两个正整数  $u_i, v_i$ ，表示第  $i$  条道路连接的两个城市的编号。 $(1 \leq u_i, v_i \leq n)$

### 输出格式

共两行，第一行表示他最多能够买多少件女装，第二行表示他买女装的最优方案的数量。即便你不能解决两个问题中的一个，你也需要输出两行答案，否则会因为格式错误而无法得分。具体的得分细则将在子任务中解释。

## 样例

### 样例输入1

```
1 | 2 1
2 | 1 2
```

### 样例输出1

```
1 | 2 1
```

### 样例解释1

- 1 | 最优方案只有一种，就是在 1 号小镇，2 号小镇各买一件女装。
- 2 | 至此，出题人已经兑现了省选与 NOI 前立下的女装 flag。

### 样例输入2

```
1 | 7 7
2 | 1 2
3 | 2 3
4 | 3 1
5 | 4 5
6 | 5 6
7 | 4 6
8 | 4 7
```

### 样例输出2

```
1 | 7
2 | 6
```

### 样例3

见附加文件。

### 样例3解释

注意，样例中除了 femaledress3.in/out 文件，还有 femaledress3.ans 文件。它的输出格式如下，第一行表示第一问的答案。第二行共  $n$  个数，第  $i$  个数表示的是把与第  $i$  小镇连通的小镇看成一个新的城市，那么第二问的答案会是多少。

这个样例，无疑是善良的出题人无私的馈赠。虽然只有一组数据，但是你可以利用 femaledress3.ans 文件，将给定的这个图拆成每个连通块，作为多组测试数据来进行测试。这个数据涵盖了测试点中所有出现性质的组合。你可以利用这个测试点，对自己的程序进行全面的检查。足量的数据组数和多种多样的数据类型，能让程序中的错误无处遁形。出题人相信，这个美妙的样例，可以给拼搏于 AC 这道题的逐梦之路上的你，提供一个有力的援助。

出题人要暂时性的离开 OI，离开信息组了。回想起回想起机房里考试失利的唏嘘，怒写 5K 代码 AC 一道题的欢喜，熬夜打 CF 的疲倦和立志板刷 BZOJ 的冲动，出题人感慨万千。但我知道，离别是为了更好的重逢。我加入信息组的这几年，感到无憾的是，我已经实现了自己的价值。如果把我的竞赛生涯比作一道 OI 题，那么我已经做出来了一个 subtask。

借这道题，借这个美妙的样例，出题人真心祝愿你能够在 NOIP、省选、NOI 的道路上走的更远。也希望这道题不会是我出的最后一道题，这段话也不会是我写给你们最后一段话。

### 限制

对于所有数据， $1 \leq n \leq 100000, 0 \leq m \leq 300000$ 。

数据编号	$n \leq$	$m \leq$	特殊限制
1	1	0	-
2-4	10	10	-
5-7	20	190	-
8-15	2000	4000	-
16	4000	4000	任意两个城市可以通过道路到达且 $m = n$
17-19	100000	100000	任意两个城市可以通过道路到达且 $m = n$
20-25	100000	300000	-

如果你答对第一问，你会得到该测试点 50% 的分数。在答对第一问的基础上答对第二问，你才能得到 100% 的分数。

保证给出的道路没有连接同一个小镇，且没有两道道路连接的小镇相同。

# Problem C. 萨温节的信封

## 题目背景

```
1 だって Happy Halloween オバケのカブだってラララ
2 今日は街中 trick or treat チョコも頂戴よ
3 だって Happy Halloween 不思議な世界へようこそ
4 疲れたジャックも歌うよ Let's trick or treat
```

众所周知，明天是萨温节。薇尔莉特又要忙起来了。

## 题目描述

众所周知，薇尔莉特是一个自动手记人偶，不过她也喜欢奔走于大街小巷之中派送信件。她喜欢写信，当然也喜欢整理信件。现在她的面前是一大堆将要送往莱顿各处的、洒上朱红色烫印的信封。马上就会有一名金黄色头发的小哥来取走这些美丽的白鸽，护送它们完成伟大的归程。

本该是这样的，但今天是战后的第一个万圣节，C·H 邮政公司突然收到了数以亿计的万圣节贺卡，堆在薇尔莉特前的早已不是美丽的白鸽，而是咯咯大笑的小恶魔。

C·H 邮政公司的小房间里有  $n$  个柜子，编号分别为  $1, 2, \dots, n$ 。薇尔莉特需要按照信封送往的街区把它们一一塞进这些柜子里。莱顿共有  $n$  个街区，因为编号连续的街区距离比较近，而今天莱顿城里的信比起往常又更加地分配不均，为了能够塞下这么多信，薇尔莉特规定，可以且只能在编号在  $i$  和  $p_i$  之间（包含  $i$  和  $p_i$  且  $i \leq p_i$ ）的柜子中放置送往第  $i$  个街区的邮件，显然在相互交叉的柜子中可能存有来自多个街区的信件。由于小哥比较懒，今天还要去泡嘉德丽雅，他只会在柜子前粗略地看一眼，并且希望可能送往第  $i$  个街区的信不少于  $b_i$  封，即如果设第  $i$  个柜子里有  $a_i$  封信，他希望  $\sum_{j=i}^{p_i} a_j \geq b_i$ ，如果第  $i$  个街区的要求没有被满足，则他会增加  $t_i$  的怨气值。

另外，薇尔莉特不会魔法，所以柜子的大小是有限制的。已知存在另外  $m$  条限制，每一条限制形如  $\sum_{j=x_i}^{y_i} a_j \leq c_i$ 。特别的是，聪明的薇尔莉特为了能够更好地安排这些信件，她事先保证了存在一个正整数  $k$ ，使得  $[x_i, y_i]$  这个区间可以恰好被分成  $k$  个形如  $[s_i, p_{s_i}]$  的区间。换句话说，存在这样的  $k$  个区间  $[s_i, p_{s_i}]$  ( $i = 1, 2, \dots, k$ ) 满足  $p_{s_i} + 1 = s_{i+1}$  且  $s_1 = x_i, p_{s_k} = y_i$ 。

当然这  $n + m$  个限制很可能不能全部被满足，现在薇尔莉特想要知道，在所有满足后  $m$  条限制的  $a_i$  中，小哥怨气值可能的最小值是多少。由于京紫的世界是一个架空世界，所以我们可以假设  $a_i$  是任意的整数（可以为负）。正因为如此，这个问题一定是有解的。（因为在只需满足后  $m$  条限制的情况下，令所有的  $a_i$  均为负数一定是一个合法的解）

## 输入格式

输入共  $n + m + 1$  行，第一行为两个整数  $n, m$ ，表示两种限制的条数。

第 2 行至第  $n + 1$  行，每行包含三个用一个空格分开的非负整数  $p_i, b_i, t_i$ ，意义同【题目描述】中所述。

第  $n + 2$  行至第  $n + m + 1$  行，每行包含三个用一个空格分开的非负整数  $x_i, y_i, c_i$ ，意义同【题目描述】中所述。

输入保证是合法的，即一定满足  $i \leq p_i, 1 \leq x_i \leq y_i \leq n$  以及薇尔莉特事先保证的一条性质。

## 输出格式

输出共 1 行，表示小哥可能的最小怨气值。

## 样例

### 样例输入1

```
1 | 5 2
2 | 1 1 1
3 | 2 2 1
4 | 3 3 1
5 | 4 4 1
6 | 5 5 1
7 | 4 5 10
8 | 1 2 1
```

### 样例输出1

```
1 | 1
```

### 样例解释1

去掉第 1 个条件或者第 2 个条件都可以满足要求。



## 样例输入2

```
1 5 3
2 3 2 1
3 3 7 1
4 5 5 1
5 5 4 4
6 5 3 3
7 1 5 7
8 2 5 7
9 3 5 4
```

## 样例输出2

```
1 3
```

## 样例解释2

一种合法的最优方案是，去掉前 3 个条件，然后令  $a_i = -100, -100, -100, 1, 3$ 。

## 样例3

见选手目录下的 *envelope/envelope3.in* 和 *envelope/envelope3.ans*，该组样例的数据范围同第 11 个测试点。

## 样例4

见选手目录下的 *envelope/envelope4.in* 和 *envelope/envelope4.ans*，该组样例的数据范围同第 21 个测试点。

## 样例5

见选手目录下的 *envelope/envelope5.in* 和 *envelope/envelope5.ans*，该组样例的数据范围同第 36 个测试点。

## 样例6

见选手目录下的 *envelope/envelope6.in* 和 *envelope/envelope6.ans*，该组样例的数据范围同第 42 个测试点。

## 限制

本题共 50 个测试点, 对于所有的测试点, 满足  $n, m \leq 5 \times 10^5, b_i, c_i, t_i \leq 10^9$ , 对于各测试点的特殊性质请参考下表:

测试点 (分值)	$n$	$m$	$b_i, c_i$	$t_i$	其他性质
1 – 5(10pts)	$\leq 2$	$\leq 2$	$\leq 10$	$\leq 10$	无
6 – 10(10pts)	$\leq 10$	$\leq 10$	$\leq 10$	$= 1$	无
11 – 14(8pts)	$\leq 2000$	$\leq 2000$	$\leq 10^9$	$= 1$	$i = p_i$
15 – 20(12pts)	$\leq 10^5$	$\leq 10^5$	$\leq 10^9$	$= 1$	$i = p_i$
21 – 26(12pts)	$\leq 10^5$	$\leq 10^5$	$\leq 10^9$	$\leq 10^9$	$i = p_i$
27 – 30(8pts)	$\leq 10^5$	$\leq 10$	$\leq 10^9$	$\leq 10^9$	无
31 – 35(10pts)	$\leq 200$	$\leq 200$	$\leq 10^9$	$= 1$	无
36 – 41(12pts)	$\leq 10^5$	$\leq 10^5$	$\leq 10^9$	$= 1$	无
42 – 46(10pts)	$\leq 10^5$	$\leq 10^5$	$\leq 10^9$	$\leq 10^9$	无
47 – 50(8pts)	$\leq 5 \times 10^5$	$\leq 5 \times 10^5$	$\leq 10^9$	$\leq 10^9$	无