

HNSDFZ第3次欢乐互测

	expression	unstring	gambler	starcraft
源文件	expression.cpp	unstring.cpp	gambler.cpp	starcraft.cpp
输入文件	expression.in	unstring.in	gambler.in	starcraft.in
输出文件	expression.out	unstring.out	gambler.out	starcraft.out
时间限制	1s	1s	1s	1s
空间限制	512MB	128MB	128MB	32MB
题目类型	传统型	传统型	传统型	传统型
出题人	riteme	Link	ruanxingzhi	Haogram

拒绝Pascal程序。

对于C/C++程序，使用以下的编译命令：

```
1 g++ source.cpp -o exec -O2 -std=c++11
```

A. 密码锁

题目描述

dyx在家里玩耍时发现了一个神奇的密码锁。然而他早已忘记了这个锁的密码，于是他随便尝试了一下，结果锁就打开了.....

锁的内部有一个很长的字符串，机智的dyx马上就发现这就是密码锁的核心。于是他研究了一下午，探寻这把锁的奥秘。

他发现这个字符串是一个表达式的形式，像下面这个样子：

1	a b&(!c^d)
---	------------

其中，每一个由小写字母组成的单词是一个变量，对应着密码锁上的一个按钮。由于按钮只能按下或不按下，于是你可以认为每个变量是一个布尔类型的（即只有true和false之分）

其余的字符就只有&、|、^、!和左右小括号。dyx发现括号是用来优先运算的，意思是这是一个会将所有变量进行计算的表达式，并且优先计算括号中的子表达式。同时，&、|、^、!都是运算符，它们分别对应的是**逻辑与**、**逻辑或**、**逻辑异或**和**逻辑非**。其中前三者运算优先级一致，当它们在同一级出现时会从左至右运算，逻辑非的优先级比它们高。当每一个变量都有相应的值时，整个表达式就会进行计算，并给出一个布尔值。dyx还发现，当整个表达式的值为true时，密码锁就会打开。

很明显，整个密码锁的输入方案共有 2^n 种，其中 n 是表达式中变量的数量。于是dyx瞬间明白为什么他一次就可以将这个密码锁解开了。然而dyx是一个勇于探究的人，他想知道到底有多少中方法可以解开一个密码锁。

不知为何，dyx又发现了一火车的密码锁。坚持不懈的dyx不停的计算着每一个密码锁能解开的方案数.....由于密码锁的表达式越来越长并且人脑计算量是 $\Theta(1)$ 的，dyx不得不需要一个程序来帮助他计算这个方案数。

输入格式

每个测试数据点有多个表达式。文件以EOF结束。

每一个表达式占一行，且中间只有小写字母和&、|、^、!、(、)。

对于&、|和^运算，它们左右会各有一个变量或子表达式。其运算规则如下：

a	b	$a \& b (a \wedge b)$
true	true	true
true	false	false
false	true	false
false	false	false

a	b	$a \mid b (a \vee b)$
true	true	true
true	false	true
false	true	true
false	false	false

a	b	$a \wedge b (a \oplus b)$
true	true	false
true	false	true
false	true	true
false	false	false

对于!运算，它后面会有一个变量或子表达式。其运算规则如下：

a	$!a (\neg a)$
true	false
false	true

输入保证表达式是有效的，且表达式中不存在缺少参数的运算符和空的括号。

为了防止此题变成不可做的NP-hard问题，输入保证表达式中每一个变量名只出现一次。

输出格式

对于每一个表达式，输出能够解开它的方案总数。由于答案可能过大，因此将答案对 $10^9 + 7$ 取模后输出。

一种方案即指对每一个变量给定一个值。

两种方案不同当且仅当至少一个变量所给定的值不同。

样例输入

1	a&b
2	a b c
3	a !(b c)

样例输出

1	1
2	7
3	5

数据范围及提示

对于第一个表达式，只有 $a = \text{true}$, $b = \text{true}$ 才为 true 。

对于第二个表达式，只要三者有一个变量为 true 就是 true 。

对于第三个表达式，先运算 $b|c$ ，然后对其取反，最后与 a 做逻辑与。

以下 n 表示变量数量， T 表示测试数据组数， L 表示表达式的总长度。

对于10%的数据， $n \leq 10$, $T = 15$ 。

对于另外10%的数据，只有逻辑或运算，没有括号。

对于另外10%的数据，不存在逻辑非和括号。

对于另外20%的数据，不存在括号。

对于另外20%的数据， $n \leq 500$ 。

对于100%的数据，变量名长度 ≤ 4 , $L \leq 10^6$, $n \leq 3 \times 10^5$ 。

其中90%的数据， $T \leq 5$ 。

B. UnString

题目描述

ZY很累。

因为最近内网的服务器地址总是改来改去。

然而ZY又非常想去捣乱。他必须知道内网的服务器的地址是啥。

但是阴险的Blue却是给地址加了个密，还TM竟然隐藏了一些字符。

但是ZY也是蛮机智的，他得到了一段被加密的地址

还有Blue傻逼丢博客的字符对照表，如果地址和博客的字符对照表相似那么ZY就知道他得到的地址是对的，虽然他仍然不能知道地址是啥。

输入格式

第一行一个字符串 S ，字符串 S 是由 $a-b$ 组成的字符串

第二行一个字符串 C ，字符串 C 是由 $a-b$ 和 $?$ 组成的字符串

输出格式

我们将 $?$ 认为是任何字符，询问 C 是否是 S 的子串

输出YES 或者 NO

样例输入

1	abc
2	a?c

样例输出

1	YES
---	-----

数据范围及提示

30% S 长度小于2000， C 长度小于20

60% S 长度小于100000， C 长度小于500

100% S 长度不超过200000, C 长度不超过2000

C. 赌徒

题目描述

HNSDFZ有一大批赌徒。他们分布在化学组、生物组、机房……这些赌场有道路相连，形成一棵树的结构。

赌场里的人越来越多。如果记某个赌场第 n 天的人数为 F_n ，则有 $F_n = 2F_{n-2} + 3F_{n-1}$ 。每个赌场的 F_1 和 F_2 是预先知道的。当然，赌徒的数量不能太多。每个赌场都会不停地清理人员，做法是把每天的人数模2333。

现在是第 m 天，3班的某个英语老师要打击赌徒。她准备打击一条链上的所有赌徒。但是她并不知道自己最多能打到多少个赌徒，所以把你捆过来算一算。

输入格式

第一行一个正整数， n 表示赌场个数， m 表示在第几天动手。

接下来 n 行，每行两个整数 F_1, F_2 。

接下来 $n - 1$ 行，每行两个整数 u, v 表示 u 和 v 有边相连。

输出格式

一行，一个正整数，表示能打击到的赌徒个数。

样例输入

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

样例输出

1	4
---	---

数据范围及提示

输入数据中：

在第一天，所有的赌场都只有1个人。

在这时打击，最多能打击到4个人。这条链是4 - 2 - 1 - 3。

$$n \leq 5 \times 10^5, m \leq 2 \times 10^9$$

D. 星际争霸

题目描述

虽然二十世纪的科技和文化进步神速，但是和後世科技和文明的跳跃式进步相比起来，也只能甘拜下风。在二十一世纪的末，人类的文明经历了巨大而且难以想像的进步。崭新的科技以难以想像的速度窜起，即使是地球上最为落後的国家也开始拥有越来越先进的电脑和资料库。在东欧共产主义解体的同时，核子武器开始变得随手可得。原来国际势力的兴衰是以资产和军事强权来作为依据，第三世界的国家却利用这个机会猛然窜起，挑战这些强权，打破国际间局势的均衡。

当系统控工学、复制和基因工程变成人人皆可掌握的科技之後，激进人道主义者和狂热的宗教团体开始质疑私人公司以基因工程图利的正当性。大众开始纷纷装置由精密工学所研制出来的人工器官，其他人则开始显现出各种各样的基因突变性状，包括了较为隐性的器官变得敏锐，到明显的心灵感应。人类基因库中所产生的这些变化，让全球各地的人本主义者感到非常的恐慌。

科技继续进步及扩张，人口增加的速度也开始飙升。在二十世纪结束的时候，地球上有着六十亿的人口。仅仅经过短短的三百年，人口暴增为两百三十亿。污染和自然资源、燃料的耗竭更是火上加油。各个国家莫不竭力寻找降低人口成长率的方法。在人口爆炸、基因突变横扫整个地球的同时，一般人都认为地球将因此而走向天灾地变的结局。

于是人类造出一艘星际巨舰，在擎天神的领导下，飞往无边的太空.....

最後，其中一艘的航舰引擎融毁了。在经过二十八年的曲速旅行之後，这些巨大的船只重新出现在三度空间中，靠近一个可居住星系的外缘。这里距离地球六万光年之远，他们的曲速引擎全毁，生命维持的系统几乎已经耗竭。所有的船只只有进入紧急状况，准备迫降在最接近的可居住行星上。

每个行星的居民努力的试图在被称为『新世界』的星球上生存。他们并不知道还有其它的同类挣扎著在别的地方求生存，只能够努力的利用手边的资源活下去。

然而，Zerg和Protoss的兵力正进攻过来，能否活下还是个问题.....

作为采矿的工具，空间建筑工程车 (Space Construction Vehicle)，简称SCV，它们的多样性和无以比拟的可靠性使得它们成为一种极有价值的建筑工具。

采矿的流程如下，然而图不能动。



SCV从指挥中心 (Command Center) 出发，移动至水晶矿处开采矿物，为了简化问题，设定为采矿不需要时间。采完矿后需要返回指挥中心将矿物送回，否则不能算采到了矿物。图片里既有正在采矿的也有才完了矿带着矿回指挥中心的。

因为地面环境复杂，所以SCV所走的路线不一定是笔直的，而是曲折的，再次你可以将其理解为以指挥中心为根的一颗有根树，而水晶矿处于该有根树的叶子节点处。有根树的每一条边都有距离，长度单位为宇宙单位。定义SCV速度为 1 宇宙单位/s。为简化问题，SCV一次采矿可以将该水晶矿采完。

有时会有一些宇宙生物（比如卡拉兽）会爬行至采矿路径上的一些点导致该路径不通。增加采矿难度。出于某种原因，地图会不断更新，矿物就会恢复原状，卡拉兽也会被更新至一个新的节点。

Zerg和Protoss的进攻已经近了，你需要尽快采更多的资源.....

输入格式

第一行有2个正整数 n, m ，表示图上有 n 个点，地图更新 m 次。

接下来 $n - 1$ 行，每行3个正整数 p, f, l ，代表描述第 p 个点，其父节点为 f ，与父节点的连边长为 l 。

接下来 x 行，有2个正整数 p, w ，表示第 p 号节点有 w 水晶矿。 x 为叶节点个数。

接下来 m 行，每行一个正整数 p, t ，表示地图更新后，有一只卡拉兽在 p 号节点位置上，Zerg和Protoss的进攻将在 t 秒后来临。如果 p 为0，表示没有卡拉兽在地图上。

输出格式

地图每次更新会将采完的矿回复原状。

输出共 m 行，每行对于一次地图更新，输出在进攻来临前采集到的最多矿物数量。

样例输入1

1	3 3
2	2 1 35
3	3 1 24
4	2 100
5	3 65
6	0 60
7	1 23
8	3 33

样例输出1

1	65
2	0
3	0

样例输入2

1	9 4
2	2 1 8
3	3 1 10
4	5 2 25
5	6 2 13
6	7 3 22
7	4 1 26
8	8 4 12
9	9 4 23
10	5 78
11	6 66
12	7 48
13	8 50
14	9 80
15	0 257
16	3 200
17	2 230
18	1 500

样例输出2

1	242
2	194
3	130
4	0

数据范围及提示

对于10%的数据: $n \leq 20, m \leq 5$

对于20%的数据: $n \leq 20, m \leq 500$

对于另外20%的数据, 地图没有卡拉兽

对于另外20%的数据, $n \leq 1000, m \leq 50$

对于100%的数据, $n \leq 1000, m \leq 500, t \leq 50000$

指挥中心的编号始终为1, 地图至多有一只卡拉兽