

NOIP 2017 模拟赛 Day1

题目	函数	堆	树
目录	function	heap	tree
源程序文件名	function.cpp	heap.cpp	tree.cpp
输入文件名	function.in	heap.in	tree.in
输出文件名	function.out	heap.out	tree.out
时间限制	2s	1s	3s
空间限制	128MB	128MB	256MB
编译命令	-O2	-O2	-O2

读入可能较大，请尽量使用读入优化

函数

一天，dt 闲着无聊，于是他定义函数 $f(n)$ 为不同的满足 $x + y = n$ 且 $\gcd(x, y) = 1$ 的有序正整数对 (x, y) 的个数，

特殊的， $f(1) = 1$ 。

dt 觉得这个函数太 naive，于是他又定义了一个函数 $g(n)$ ，其中

$$g(n) = \sum_{d|n} f\left(\left\lfloor \frac{n}{d} \right\rfloor\right)$$

dt 还是觉得这个函数太 simple，于是他定义了一个 **复合函数** $F_k(n)$ ，其中

$$F_k(n) = \begin{cases} n & \text{if } k = 0 \\ f(F_{k-1}(n)) & \text{if } k > 0, k \equiv 1 \pmod{2} \\ g(F_{k-1}(n)) & \text{if } k > 0, k \equiv 0 \pmod{2} \end{cases}$$

dt 觉得这个函数还是很 simple，不过他懒得求这个函数的值了，就把这个问题丢给了你。

输入格式

第一行一个整数 T ($T \leq 5$)，表示数据组数。

之后 T 行，每行两个整数 n, k 。

输出格式

输出 T 行，每行一个整数表示 $F_k(n) \bmod 1000000007$

样例

输入

```
3
15 2
233 3
233333333333 3
```

输出

8

112

271986803

数据范围

对于50% 的数据, $n \leq 10^6, k \leq 10$

对于100% 的数据, $1 \leq n, k \leq 10^{12}$

堆

请注意可能的 `int` 溢出。

这里的堆是一个无限大的满二叉树, 根节点的编号为1, 编号为 x 的节点的左右儿子分别编号为 $2x, 2x + 1$ 。

一次行走可以由一个只含有 L、R、P 的字符串表示, 从根开始, L表示走到左儿子, R表示走到右儿子, P表示暂停一步。

定义一次行走的值为这次行走的终点的编号。

比如 LPR代表 $1 \rightarrow 2 \rightarrow 2 \rightarrow 5$ 这次行走, 其值为5

对于一个只含有 L、R、P、* 的字符串, 其中 *可以匹配 L、R、P三个字符中任意一个,

你的任务是计算这个字符串能匹配的所有行走的值的和 $\text{mod } 1000000007$ 。

比如 $L \star R$, 能匹配 3 个行走: LLR, LRR 和 LPR, 值分别是9、11、5, 和为25

相同的值可以算多次, 比如 PL和 LP可以同时被算进 $\star \star$ 的和里。

不过事情还没完, 我们将额外输入一个整数 k , 设输入的字符串为 s , 你需要将 k 个 s 依次连接之后对于得到的字符串计算结果。

输入格式

第一行一个字符串 s

第二行一个整数 k

输出格式

输出一个整数表示答案

样例 1

输入

$L \star R$

1

输出

25

样例 2

输入

*L*P*R

1000

输出

502025683

数据范围

对于20% 的数据，s 中最多有3 个 *， $k = 1$

对于50% 的数据， $k = 1$

对于100% 的数据，s 的长度 $\leq 10^5$ ， $1 \leq k \leq 10^{18}$

树

本题开栈

给你一棵边带权的树，定义一条路径的长度为这条路径上所有边权的最大值，你要支持两个操作：

查询一个点走到一条路径上的最短路

添加一个编号为当前点数+1 的点和一条边，保证之后的图仍是树。

输入格式

第一行两个整数 n, q ，表示初始有 n 个点，共有 q 个操作。

之后 $n-1$ 行，每行三个整数 u, v, a 表示初始有一条从 u 到 v ，长度为 a 的边。

之后 q 行，每行第一个整数 op 表示操作类型。

如果 $op = 1$ 后面有三个整数 a, u, v ，表示查询从 a 走到 (u, v) 路径上的最短路。

否则 $op = 2$ 后面有两个整数 a, b ，表示添加一个编号为当前点数+1的点和一条从这个点到 a 长度为 b 的边。

除了 op 以外，这 q 行输入的所有数均要异或上一次询问的答案，初始为0

输出格式

对于每个 $op = 1$ ，输出一行一个整数表示答案，如果这个点已经在这条路径上则输出0。

样例

输入

```
10 10
10 1 237837066
7 4 333101381
```

```
5 6 764410644
2 9 800406055
2 3 977449294
10 9 85077389
6 8 883423879
7 5 483282546
3 4 109819496
2 2 237524075
1 7 3 2
2 333101376 243353439
2 333101391 970164111
2 333101389 489066126
1 333101377 333101376 333101383
2 5 947013122
2 15 885096691
1 7 9 10
1 977449284 977449286 977449291
```

输出

```
333101381
0
977449294
977449294
```

数据范围

对于前10% 的数据， $1 \leq n, q \leq 20$ 。

对于前30% 的数据， $1 \leq n, q \leq 2000$ 。

对于另20% 的数据，所有边满足 $v = u + 1$ ，且 $op \neq 2$ 。

对于另20% 的数据， $op \neq 2$ 。

对于100% 的数据， $1 \leq n, q \leq 2 * 10^5$ ，解密后数据均合法，且所有边权为不超过 10^9 的自然数。