

第 34 届全国青少年信息学奥林匹克竞赛

CCF NOI 2017

第一试

时间：2017 年 7 月 19 日 08:00 ~ 13:00

题目名称	整数	蚯蚓排队	泳池
题目类型	传统型	传统型	传统型
目录	integer	queue	pool
可执行文件名	integer	queue	pool
输入文件名	integer.in	queue.in	pool.in
输出文件名	integer.out	queue.out	pool.out
每个测试点时限	2.0 秒	2.0 秒	3.0 秒
内存限制	512 MB	2 GB	512 MB
测试点数目	25	25	20
每个测试点分值	4	4	5

提交源程序文件名

对于 C++ 语言	integer.cpp	queue.cpp	pool.cpp
对于 C 语言	integer.c	queue.c	pool.c
对于 Pascal 语言	integer.pas	queue.pas	pool.pas

编译选项

对于 C++ 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 C 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 Pascal 语言	-O2	-O2	-O2

整数 (integer)

【题目背景】

在人类智慧的山巅，有着一台字长为 1048576 位的超级计算机，著名理论计算机科学家 P 博士正用它进行各种研究。不幸的是，这天台风切断了电力系统，超级计算机无法工作，而 P 博士明天就要交实验结果了，只好求助于学过 OI 的你.....

【题目描述】

P 博士将他的计算任务抽象为对一个整数的操作。

具体来说，有一个整数 x ，一开始为 0。

接下来有 n 个操作，每个操作都是以下两种类型中的一种：

- **1 $a\ b$** ：将 x 加上整数 $a \cdot 2^b$ ，其中 a 为一个整数， b 为一个非负整数
- **2 k** ：询问 x 在用二进制表示时，位权为 2^k 的位的值（即这一位上的 1 代表 2^k ）
保证在任何时候， $x \geq 0$ 。

【输入格式】

从文件 *integer.in* 中读入数据。

输入的第一行包含四个正整数 n, t_1, t_2, t_3 ， n 的含义见题目描述， t_1, t_2, t_3 的具体含义见子任务。

接下来 n 行，每行给出一个操作，具体格式和含义见题目描述。

同一行输入的相邻两个元素之间，用恰好一个空格隔开。

【输出格式】

输出到文件 *integer.out* 中。

对于每个询问操作，输出一行，表示该询问的答案（0 或 1）。对于加法操作，没有任何输出。

【样例 1 输入】

```
10 3 1 2
1 100 0
1 2333 0
1 -233 0
2 5
2 7
```

```
2 15
1 5 15
2 15
1 -1 12
2 15
```

【样例 1 输出】

```
0
1
0
1
0
```

【样例 1 解释】

样例中有 10 个操作：第 1 个为将 x 加上 100×2^0 ，操作后， $x = 100$ ；
第 2 个为将 x 加上 2333×2^0 ，操作后， $x = 2433$ ；
第 3 个为将 x 加上 -233×2^0 ，操作后， $x = 2200$ ；
第 4 个为询问 x 位权为 2^5 的位上的值， x 在二进制下为 100010011000，答案为 0；
第 5 个为询问 x 位权为 2^7 的位上的值， x 在二进制下为 100010011000，答案为 1；
第 6 个为询问 x 位权为 2^{15} 的位上的值， x 在二进制下为 100010011000，答案为 0；
第 7 个为将 x 加上 $5 \times 2^{15} = 163840$ ，操作后， $x = 166040$ ；
第 8 个为询问 x 位权为 2^{15} 的位上的值， x 在二进制下为 101000100010011000，答案为 1；
第 9 个为将 x 加上 $-1 \times 2^{12} = -4096$ ，操作后， $x = 161944$ ；
第 10 个为询问 x 位权为 2^{15} 的位上的值， x 在二进制下为 100111100010011000，答案为 0。

【样例 2】

见选手目录下的 *integer/integer2.in* 与 *integer/integer2.ans*。该组样例的数据范围同第 7 个测试点。

【样例 3】

见选手目录下的 *integer/integer3.in* 与 *integer/integer3.ans*。该组样例的数据范围同第 13 个测试点。

【样例 4】

见选手目录下的 *integer/integer4.in* 与 *integer/integer4.ans*。该组样例的数据范围同第 14 个测试点。

【子任务】

在所有测试点中, $1 \leq t_1 \leq 3, 1 \leq t_2 \leq 4, 1 \leq t_3 \leq 2$ 。不同的 t_1, t_2, t_3 对应的特殊限制如下:

- 对于 $t_1 = 1$ 的测试点, 满足 $a = 1$
- 对于 $t_1 = 2$ 的测试点, 满足 $|a| = 1$
- 对于 $t_1 = 3$ 的测试点, 满足 $|a| \leq 10^9$
- 对于 $t_2 = 1$ 的测试点, 满足 $0 \leq b, k \leq 30$
- 对于 $t_2 = 2$ 的测试点, 满足 $0 \leq b, k \leq 100$
- 对于 $t_2 = 3$ 的测试点, 满足 $0 \leq b, k \leq n$
- 对于 $t_2 = 4$ 的测试点, 满足 $0 \leq b, k \leq 30n$
- 对于 $t_3 = 1$ 的测试点, 保证所有询问操作都在所有修改操作之后
- 对于 $t_3 = 2$ 的测试点, 不保证询问操作和修改操作的先后顺序

本题共 25 个测试点, 每个测试点 4 分。各个测试点的数据范围如下:

测试点编号	$n \leq$	t_1	t_2	t_3	
1	10	3	1	2	
2	100		2		
3	2000				
4	4000	1		3	1
5	6000	3			
6	8000	2			
7	9000	3	4	2	
8	10000		3		
9	30000		4		
10	50000		3	1	
11	60000			3	2
12	65000	2		4	
13	70000	3			
14	200000				
15	300000	2			
16	400000	3			
17	500000		3		
18	600000		4		
19	700000				
20	800000	1			
21	900000	2	3	1	
22	930000	3			3
23	960000		4		
24	990000		3		2
25	1000000		4		

蚯蚓排队 (queue)

【题目描述】

蚯蚓幼儿园有 n 只蚯蚓。幼儿园园长神刀手为了管理方便，时常让这些蚯蚓们列队表演。

所有蚯蚓用从 1 到 n 的连续正整数编号。每只蚯蚓的长度可以用一个正整数表示，根据入园要求，所有蚯蚓的长度都不超过 6。神刀手希望这些蚯蚓排成若干个队伍，初始时，每只蚯蚓各自排成一个仅有一只蚯蚓的队伍，该蚯蚓既在队首，也在队尾。

神刀手将会依次进行 m 次操作，每个操作都是以下三种操作中的一种：

1. 给出 i 和 j ，令 i 号蚯蚓与 j 号蚯蚓所在的两个队伍合并为一个队伍，具体来说，令 j 号蚯蚓紧挨在 i 号蚯蚓之后，其余蚯蚓保持队伍的前后关系不变。
2. 给出 i ，令 i 号蚯蚓与紧挨其后的一只蚯蚓分离为两个队伍，具体来说，在分离之后， i 号蚯蚓在其中一个队伍的队尾，原本紧挨其后的那一只蚯蚓在另一个队伍的队首，其余蚯蚓保持队伍的前后关系不变。
3. 给出一个正整数 k 和一个长度至少为 k 的数字串 s ，对于 s 的每个长度为 k 的连续子串 t （这样的子串共有 $|s|-k+1$ 个，其中 $|s|$ 为 s 的长度），定义函数 $f(t)$ ，询问所有这些 $f(t)$ 的乘积对 998244353 取模后的结果。其中 $f(t)$ 的定义如下：

对于当前的蚯蚓队伍，定义某个蚯蚓的向后 k 数字串为：从该蚯蚓出发，沿队伍的向后方向，寻找最近的 k 只蚯蚓（包括其自身），将这些蚯蚓的长度视作字符连接而成的数字串；如果这样找到的蚯蚓不足 k 只，则其没有向后 k 数字串。例如蚯蚓的队伍为 10 号蚯蚓在队首，其后是 22 号蚯蚓，其后是 3 号蚯蚓（为队尾），这些蚯蚓的长度分别为 4、5、6，则 10 号蚯蚓的向后 3 数字串为 456，22 号蚯蚓没有向后 3 数字串，但其向后 2 数字串为 56，其向后 1 数字串为 5。

而 $f(t)$ 表示所有蚯蚓中，向后 k 数字串恰好为 t 的蚯蚓只数。

【输入格式】

从文件 `queue.in` 中读入数据。

输入文件的第一行有两个正整数 n, m ，分别表示蚯蚓的只数与操作次数。

第二行包含 n 个不超过 6 的正整数，依次表示编号为 $1, 2, \dots, n$ 的蚯蚓的长度。

接下来 m 行，每行表示一个操作。每个操作的格式可以为：

- 1 $i\ j$ ($1 \leq i, j \leq n$) 表示：令 i 号与 j 号蚯蚓所在的两个队伍合并为一个队伍，新队伍中， j 号蚯蚓紧挨在 i 号蚯蚓之后。保证在此操作之前， i 号蚯蚓在某个队伍的队尾， j 号蚯蚓在某个队伍的队首，且两只蚯蚓不在同一个队伍中。
- 2 i ($1 \leq i \leq n$) 表示：令 i 号蚯蚓与紧挨其后一个蚯蚓分离为两个队伍。保证在此操作之前， i 号蚯蚓不是某个队伍的队尾。

- $\underline{3} s k$ (k 为正整数, s 为一个长度至少为 k 的数字串) 表示: 询问 s 的每个长度为 k 的子串 t 的 $f(t)$ 的乘积, 对 998244353 取模的结果。 $f(t)$ 的定义见题目描述。

同一行输入的相邻两个元素之间, 用恰好一个空格隔开。

输入文件可能较大, 请不要使用过于缓慢的读入方式。

【输出格式】

输出到文件 `queue.out` 中。

依次对于每个形如 $\underline{3} s k$ 的操作, 输出一行, 仅包含一个整数, 表示询问的结果。

【样例 1 输入】

```
5 9
3 1 3 5 3
3 333135 2
3 333135 1
1 1 3
1 2 5
1 3 2
1 5 4
3 333135 2
3 333135 1
3 333135 3
```

【样例 1 输出】

```
0
81
1
81
0
```

【样例 1 解释】

第一次询问: 由于每个队伍均只有一只蚯蚓, 所以没有任何蚯蚓有向后 2 数字串, 答案为 $f(\underline{33}) \times f(\underline{33}) \times f(\underline{31}) \times f(\underline{13}) \times f(\underline{35}) = 0 \times 0 \times 0 \times 0 \times 0 = 0$ 。

第二次询问: 每个队伍仍只有一只蚯蚓, 每只蚯蚓的向后 1 数字串就是将自己的长度视为字符的数字串, 即: 得到的 5 个向后 1 数字串为 1、3、3、3、5 (不分先后顺

【样例 2 解释】

对于第四次、第七次询问，输入的 s 为 30 个字符 1，所有 $f(t)$ 的乘积是 $2^{30} = 1073741824$ ，输出的结果是这个数对于 998244353 取模的结果。

【样例 3】

见选手目录下的 `queue/queue3.in` 与 `queue/queue3.ans`。
该组样例的数据范围同第 5 个测试点。

【样例 4】

见选手目录下的 `queue/queue4.in` 与 `queue/queue4.ans`。
该组样例的数据范围同第 10 个测试点。

【样例 5】

见选手目录下的 `queue/queue5.in` 与 `queue/queue5.ans`。
该组样例的数据范围同第 15 个测试点。

【样例 6】

见选手目录下的 `queue/queue6.in` 与 `queue/queue6.ans`。
该组样例的数据范围同第 20 个测试点。

【子任务】

保证 $n \leq 2 \times 10^5$ ， $m \leq 5 \times 10^5$ ， $k \leq 50$ 。

设 $\sum |s|$ 为某个输入文件中所有询问的 s 的长度总和，则 $\sum |s| \leq 10^7$ 。

设 c 为某个输入文件中形如 2 i 的操作的次数，则 $c \leq 10^3$ 。

每个测试点的详细信息见下表：

测试点编号	n	m	k	$\sum s $	c	全为 <u>1</u>	
1	$= 1$	$\leq 10^3$	$= 1$	$\leq 10^3$	$= 0$	No	
2	≤ 20	≤ 40	≤ 10				
3	≤ 150	$\leq 2,000$	≤ 50		$\leq 10^3$		
4	≤ 500	≤ 600			$= 0$		
5	$\leq 10^3$	$\leq 2,000$			$\leq 10^3$		
6	$\leq 5 \times 10^4$	$\leq 6 \times 10^4$	≤ 5	$\leq 5 \times 10^4$	$= 0$	Yes	
7			≤ 50				No
8							
9				$\leq 10^3$			
10			$\leq 8 \times 10^4$	$\leq 2.5 \times 10^6$	$= 0$		
11			$\leq 10^3$				
12	$\leq 10^5$	$\leq 1.1 \times 10^5$	≤ 6		$\leq 10^5$	$= 0$	Yes
13			≤ 50				No
14						$\leq 10^3$	
15				$= 0$			
16			$\leq 1.5 \times 10^5$	$\leq 5 \times 10^6$	$\leq 10^3$		
17			$\leq 10^3$				
18	$\leq 2 \times 10^5$	$\leq 5 \times 10^5$	$= 1$		$\leq 10^7$	$= 0$	
19					$\leq 10^3$		
20		$\leq 2.5 \times 10^5$	≤ 7	$\leq 2 \times 10^5$	$= 0$	Yes	
21			≤ 50				No
22						$\leq 10^3$	
23				$= 0$			
24			$\leq 3 \times 10^5$	$\leq 10^7$	$\leq 10^3$		
25					$\leq 10^3$		

如果一个测试点的“全为 1”的一列为“Yes”，表示该测试点的所有蚯蚓的长度均为 1，并且所有询问串 s 的每一位也均为 1。

泳池 (pool)

【题目描述】

久莲是个爱玩的女孩子。

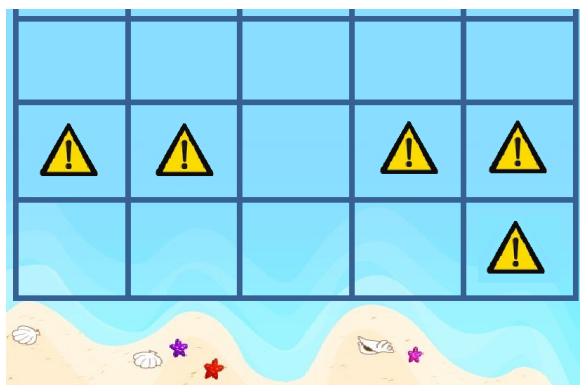
暑假终于到了，久莲决定请她的朋友们来游泳，她打算先在她家的私人海滩外圈一块长方形的海域作为游泳场。然而大海里有着各种各样的危险，有些地方水太深，有些地方有带毒的水母出没。她想让圈出来的这一块海域都是安全的。

经过初步的分析，她把这块海域抽象成了一个底边长为 N 米，高为 1001 米的长方形网格。其中网格的底边对应着她家的私人海滩，每一个 1×1 的小正方形都代表着一个单位海域。她拜托了她爸爸明天去测量每一个小正方形是否安全。在得知了信息之后，她要做的就是圈出她想要的游泳场啦。

她心目中理想的游泳场满足如下三个条件：

- 必须保证安全性。即游泳场中的每一个单位海域都是安全的。
- 必须是矩形。即游泳场必须是整个网格中的一个 $a \times b$ 的子网格。
- 必须和海滩相邻。即游泳场的下边界必须紧贴网格的下边界。

例如：当 $N = 5$ 时，若测量的结果如下（因为 1001 太大，这儿只画出网格最下面三行的信息，其他部分都是危险的）。



那么她可以选取最下面一行的 1×4 的子海域，也可以选择第三列的 3×1 的子海域。注意她不能选取最上面一行的 1×5 的子海域，因为它没有与海滩相邻。

为了让朋友们玩的开心，她想让游泳场的面积尽可能的大。因此她会选取最下面那一行的 1×4 的子海域作为最终方案。

虽然她要明天才能知道每一个单位海域是否安全，但是她现在就想行动起来估计一下她的游泳场面积有多大。经过简单的估计，她假设每一个单位海域都有独立的 q 的概率是安全的， $1 - q$ 的概率是不安全的。她想要知道她能选择的最大的游泳场的面积恰好为 K 的概率是多少。

然而久莲对数学并不感兴趣，因此她想让你来帮她计算一下这个数值。

【输入格式】

从文件 *pool.in* 中读入数据。

输入一行四个正整数 N, K, x, y , 其中 $1 \leq x < y < 998244353$ 。 q 的取值为 $\frac{x}{y}$ 。

【输出格式】

输出到文件 *pool.out* 中。

输出一行一个整数表示答案在模 998244353 意义下的取值。

即设答案化为最简分式后的形式为 $\frac{a}{b}$, 其中 a 和 b 的互质。输出整数 x 使得 $bx \equiv a \pmod{998244353}$ 且 $0 \leq x < 998244353$ 。可以证明这样的整数 x 是唯一的。

【样例 1 输入】

10 5 1 2

【样例 1 输出】

342025319

【样例 2】

见选手目录下的 *pool/pool2.in* 与 *pool/pool2.ans*。

【样例 3】

见选手目录下的 *pool/pool3.in* 与 *pool/pool3.ans*。

【提示】

$x^{p-1} \equiv 1 \pmod{p}$, 其中 p 为质数, $x \in [1, p)$ 。

【子任务】

测试点编号	N	K
1,2	$= 1$	≤ 1000
3	≤ 10	≤ 8
4	≤ 10	≤ 9
5	≤ 10	≤ 10
6	≤ 1000	≤ 7
7	≤ 1000	≤ 8
8	≤ 1000	≤ 9
9,10,11	≤ 1000	≤ 100
12,13,14	≤ 1000	≤ 1000
15,16	$\leq 10^9$	≤ 10
17,18	$\leq 10^9$	≤ 100
19,20	$\leq 10^9$	≤ 1000