Problem A. 词典

Time limit 1000 ms

Mem limit 524288 kB

题目描述

小 S 的词典里有 n 个两两不同的、长度均为 m 的单词 $w_1, w_2, \cdots, w_n$。每个单词都是一个小写字母构成的字符串。

小 S 可以做以下操作任意多次（可以不做）：选择词典中的任意一个单词，交换其中任意两个字符。

对于每个 $1 \leq i \leq n$，小 S 想知道，是否可以通过以上操作得到新的 n 个单词 $w'_1, w'_2, \cdots, w'_n$，使得对于每个 $j \neq i$，w'_i 的字典序比 w'_j 都要小。**对于 $n = 1$ 的情况，我们约定：上述性质是自然成立的。**

对于两个同样长度的字符串 $s = s_1 s_2 \cdots s_L$ 和 $t = t_1 t_2 \cdots t_L$，称字符串 s 字典序小于字符串 t，当且仅当以下条件成立：存在位置 i，在第 i 个字符之前 s 和 t 都相同，而且 $s_i < t_i$，即小写字母 s_i 在英文字母顺序中先于 t_i。

输入格式

从文件 `dict.in` 中读入数据。

输入的第一行包含两个正整数 n 和 m，分别表示单词个数和单词长度。

接下来 n 行，每行包含一个长度为 m 的小写字母字符串 w_i， 表示一个单词。

输出格式

输出到文件 `dict.out` 中。

输出一行，其中包含一个长度为 n 的 `01` 字符串 a；对于 $1 \leq i \leq n$，如果题目描述中的性质成立，则 $a_i = 1$，否则 $a_i = 0$。

样例 1

Input	Output
4 7 abandon bananaa baannaa notnotn	1110

- 不做任何操作，第一个单词字典序最小，因此输出第一个字符为 `1`； - 交换 `bananaa` 的前两个字符以及 `abandon` 的第三个和第六个字符，得到 `abondan`, `abnanaa`, `baannaa`, `notnotn`，此时第二个单词字典序最小，因此输出第二个字符为 `1`； - 交换 `baannaa` 的第一个和最后一个字符得到 `aaannab`，其余字符串不变，此时第三个单词字典序最小，因此输出第三个字符为 `1`； - 无论如何操作，第四个单词不会小于第二个单词，因此输出第四个字符为 `0`。

样例 2

见附加文件中的 [`dict2.in`](file:dict2.in) 和 [`dict2.ans`](file:dict2.ans)。

该组样例满足测试点 4 的限制。

样例 3

见附加文件中的 [`dict3.in`](file:dict3.in) 和 [`dict3.ans`](file:dict3.ans)。

该组样例满足测试点 7 的限制。

样例 4

见附加文件中的 [`dict4.in`](file:dict4.in) 和 [`dict4.ans`](file:dict4.ans)。

该组样例满足测试点 10 的限制。

数据范围与提示

对于所有测试数据，保证：$1 \le n \le 3\ 000$，$1 \le m \le 3\ 000$，w_i 为长度为 m 的小写字母字符串且两两不同。

测试点编号	$n \le$	$m \le$
1	1	1
$2 \sim 4$	26	1

测试点编号	$n \le$	$m \le$
$5 \sim 7$	15	2
8	300	300
9	10^3	10^3
10	$3\ 000$	$3\ 000$

Problem B. 三值逻辑

Time limit 1000 ms

Mem limit 524288 kB

题目描述

小 L 今天学习了 Kleene 三值逻辑。

在三值逻辑中，一个变量的值可能为：真（$True$，简写作 T）、假（$False$，简写作 F）或未确定（$Unknown$，简写作 U）。

在三值逻辑上也可以定义逻辑运算。由于小 L 学习进度很慢，只掌握了逻辑非运算 $\neg$，其运算法则为：

$$\neg T = F, \neg F = T, \neg U = U.$$

现在小 L 有 n 个三值逻辑变量 $x_1, \cdots, x_n$。小 L 想进行一些有趣的尝试，于是他写下了 m 条语句。语句有以下三种类型，其中 $\leftarrow$ 表示赋值：

1. $x_i \leftarrow v$，其中 v 为 T, F, U 的一种；
2. $x_i \leftarrow x_j$；
3. $x_i \leftarrow \neg x_j$。

一开始，小 L 会给这些变量赋初值，然后按顺序运行这 m 条语句。

小 L 希望执行了所有语句后，所有变量的最终值与初值都相等。在此前提下，小 L 希望初值中 $Unknown$ 的变量尽可能少。

在本题中，你需要帮助小 L 找到 $Unknown$ 变量个数最少的赋初值方案，使得执行了所有语句后所有变量的最终值和初始值相等。小 L 保证，至少对于本题的所有测试用例，这样的赋初值方案都必然是存在的。

输入格式

从文件 `tribool.in` 中读入数据。

本题的测试点包含有多组测试数据。

输入的第一行包含两个整数 c 和 t，分别表示测试点编号和测试数据组数。对于样例，c 表示该样例与测试点 c 拥有相同的限制条件。

接下来，对于每组测试数据：

- 输入的第一行包含两个整数 n 和 m，分别表示变量个数和语句条数。
- 接下来 m 行，按运行顺序给出每条语句。
    - 输入的第一个字符 v 描述这条语句的类型。保证 v 为 `TFU+-` 的其中一种。
    - 若 v 为 `TFU` 的某一种时，接下来给出一个整数 i，表示该语句为 $x_i \leftarrow v$；
    - 若 v 为 `+`，接下来给出两个整数 i, j，表示该语句为 $x_i \leftarrow x_j$；
    - 若 v 为 `-`，接下来给出两个整数 i, j，表示该语句为 $x_i \leftarrow \neg x_j$。

输出格式

输出到文件 `tribool.out` 中。

对于每组测试数据输出一行一个整数，表示所有符合条件的赋初值方案中，$Unknown$ 变量个数的最小值。

样例 1

Input	Output
1 3 3 3 - 2 1 - 3 2 + 1 3 3 3 - 2 1 - 3 2 - 1 3 2 2 T 2 U 2	0 3 1

第一组测试数据中，m 行语句依次为

- $x_2 \leftarrow \neg x_1$；
- $x_3 \leftarrow \neg x_2$；
- $x_1 \leftarrow x_3$。

一组合法的赋初值方案为 $x_1 = T, x_2 = F, x_3 = T$，共有 0 个 $Unknown$ 变量。因为不存在赋初值方案中有小于 0 个 $Unknown$ 变量，故输出为 0。

第二组测试数据中，m 行语句依次为

- $x_2 \leftarrow \neg x_1$；
- $x_3 \leftarrow \neg x_2$；
- $x_1 \leftarrow \neg x_3$。

唯一的赋初值方案为 $x_1 = x_2 = x_3 = U$，共有 3 个 $Unknown$ 变量，故输出为 3。

第三组测试数据中，m 行语句依次为

- $x_2 \leftarrow T$；
- $x_2 \leftarrow U$；

一个最小化 $Unknown$ 变量个数的赋初值方案为 $x_1 = T, x_2 = U$。$x_1 = x_2 = U$ 也是一个合法的方案，但它没有最小化 $Unknown$ 变量的个数。

样例 2

见附加文件中的 [`tribool2.in`](file:tribool2.in) 和 [`tribool2.ans`](file:tribool2.ans)。

该组样例满足测试点 2 的限制。

样例 3

见附加文件中的 [`tribool3.in`](file:tribool3.in) 和 [`tribool3.ans`](file:tribool3.ans)。

该组样例满足测试点 5 的限制。

样例 4

见附加文件中的 [`tribool4.in`](file:tribool4.in) 和 [`tribool4.ans`](file:tribool4.ans)。

该组样例满足测试点 8 的限制。

数据范围与提示

对于所有测试数据，保证：

- $1 \leq t \leq 6$，$1 \leq n, m \leq 10^5$；
- 对于每个操作，v 为 `TFU+-` 中的某个字符，$1 \leq i, j \leq n$。

测试点编号	$n, m \leq$	v 可能的取值
$1, 2$	10	$TFU+-$
3	10^3	TFU

测试点编号	$n, m \leq$	v 可能的取值
4	10^5	TFU
5	10^3	$U+$
6	10^5	$U+$
7	10^3	$+-$
8	10^5	$+-$
9	10^3	$TFU+-$
10	10^5	$TFU+-$

Problem C. 双序列拓展

Time limit 1000 ms

Mem limit 524288 kB

题目描述

称某个序列 $B = \{b_1, b_2, \cdots, b_n\}$ 是另一个序列 $A = \{a_1, a_2, \cdots, a_m\}$ 的**拓展**当且仅当存在**正整数**序列 $L = \{l_1, l_2, \cdots, l_m\}$，将 a_i 替换为 l_i 个 a_i 后得到序列 B。例如，

- $\{1, 3, 3, 3, 2, 2, 2\}$ 是 $\{1, 3, 3, 2\}$ 的拓展，取 $L = \{1, 1, 2, 3\}$ 或 $\{1, 2, 1, 3\}$；
- 而 $\{1, 3, 3, 2\}$ 不是 $\{1, 3, 3, 3, 2\}$ 的拓展，$\{1, 2, 3\}$ 不是 $\{1, 3, 2\}$ 的拓展。

小 R 给了你两个序列 X 和 Y，他希望你找到 X 的一个长度为 $l_0 = 10^{100}$ 的拓展 $F = \{f_i\}$ 以及 Y 的一个长度为 l_0 的拓展 $G = \{g_i\}$，使得任意 $1 \le i, j \le l_0$ 都有 $(f_i - g_i)(f_j - g_j) > 0$。由于序列太长，你只需要告诉小 R 是否存在这样的两个序列即可。

为了避免你扔硬币蒙混过关，小 R 还给了 q 次额外询问，每次额外询问中小 R 会修改 X 和 Y 中若干元素的值。你需要对每次得到的新的 X 和 Y 都进行上述的判断。

询问之间是独立的，每次询问中涉及的修改均在原始序列上完成。

输入格式

从文件 `expand.in` 中读入数据。

输入的第一行包含四个整数 c, n, m, q，分别表示测试点编号、序列 X 的长度、序列 Y 的长度和额外询问的个数。对于样例，c 表示该样例与测试点 c 拥有相同的限制条件。

输入的第二行包含 n 个整数 $x_1, x_2, \cdots, x_n$，描述序列 X。

输入的第三行包含 m 个整数 $y_1, y_2, \cdots, y_m$，描述序列 Y。

接下来依次描述 q 组额外询问。对于每组额外询问：

- 输入的第一行包含两个整数 k_x 和 k_y，分别表示对序列 X 和 Y 产生的修改个数。
- 接下来 k_x 行每行包含两个整数 p_x, v_x，表示将 x_{p_x} 修改为 v_x。
- 接下来 k_y 行每行包含两个整数 p_y, v_y，表示将 y_{p_y} 修改为 v_y。

输出格式

输出到文件 `expand.out` 中。

输出一行，其中包含一个长度为 $(q+1)$ 的 `01` 序列，序列的第一个元素表示初始询问的答案，之后 q 个元素依次表示每组额外询问的答案。对于每个询问，如果存在满足题目条件的序列 F 和 G，输出 `1`，否则输出 `0`。

样例 1

Input	Output
3 3 3 3 8 6 9 1 7 4 1 0 3 0 0 2 1 8 3 5 1 1 2 8 1 7	1001

由于 F 和 G 太长，用省略号表示重复最后一个元素直到序列长度为 l_0。如 $\{1,2,3,3,\cdots\}$ 表示序列从第三个元素之后都是 3。

以下依次描述四次询问，其中第一次询问为初始询问，之后的三次为额外询问：

1. $A=\{8,6,9\}$，$B=\{1,7,4\}$，取 $F=\{8,8,6,9,\cdots\},G=\{1,7,4,4,\cdots\}$；
2. $A=\{8,6,0\}$，$B=\{1,7,4\}$，可以证明不存在满足要求的方案；
3. $A=\{8,6,9\}$，$B=\{8,7,5\}$，可以证明不存在满足要求的方案；
4. $A=\{8,8,9\}$，$B=\{7,7,4\}$，取 $F=\{8,8,9,\cdots\},G=\{7,7,4,\cdots\}$。

样例 2

见附加文件中的 [`expand2.in`](file:expand2.in) 和 [`expand2.ans`](file:expand2.ans)。

该组样例满足测试点 4 的限制。

样例 3

见附加文件中的 [`expand3.in`](file:expand3.in) 和 [`expand3.ans`](file:expand3.ans)。

该组样例满足测试点 7 的限制。

样例 4

见附加文件中的 [`expand4.in`](file:expand4.in) 和 [`expand4.ans`](file:expand4.ans)。

该组样例满足测试点 9 的限制。

样例 5

见附加文件中的 [`expand5.in`](file:expand5.in) 和 [`expand5.ans`](file:expand5.ans)。

该组样例满足测试点 18 的限制。

数据范围与提示

对于所有测试数据，保证：

- $1 \le n, m \le 5 \times 10^5$；
- $0 \le q \le 60$；
- $0 \le x_i, y_i < 10^9$；
- $0 \le k_x, k_y \le 5 \times 10^5$，且所有额外询问的 $(k_x + k_y)$ 的和不超过 5×10^5；
- $1 \le p_x \le n$，$1 \le p_y \le m$，$0 \le v_x, v_y < 10^9$；
- 对于每组额外询问，p_x 两两不同，p_y 两两不同。

测试点编号	$n, m \le$	特殊性质
1	1	否
2	2	否
$3, 4$	6	否
5	200	否
$6, 7$	2000	否
$8, 9$	4×10^4	是
$10, 11$	1.5×10^5	是
$12 \sim 14$	5×10^5	是
$15, 16$	4×10^4	否
$17, 18$	1.5×10^5	否

测试点编号	$n, m \leq$	**特殊性质**
$19, 20$	5×10^5	否

特殊性质：对于每组询问（包括初始询问和额外询问），保证 $x_1 < y_1$，且 x_n 是序列 X 唯一的一个最小值，y_m 是序列 Y 唯一的一个最大值。

Problem D. 天天爱打卡

Time limit 2000 ms
Mem limit 524288 kB

题目描述

小 T 同学非常热衷于跑步。为了让跑步更加有趣，他决定制作一款叫做《天天爱打卡》的软件，使得用户每天都可以进行跑步打卡。

开发完成后，小 T 同学计划进行试运行，他找了大 Y 同学来帮忙。试运行共 n 天，编号为从 1 到 n。

对大 Y 同学来说，如果某天他选择跑步打卡，那么他的能量值会减少 d。初始时，他的能量值是 0，并且试运行期间他的**能量值可以是负数**。

而且大 Y 不会**连续**跑步打卡**超过** k 天；即不能存在 $1 \leq x \leq n-k$，使得他在第 x 到第 $x+k$ 天均进行了跑步打卡。

小 T 同学在软件中设计了 m 个挑战，第 i（$1 \leq i \leq m$）个挑战可以用三个正整数 (x_i, y_i, v_i) 描述，表示如果在第 x_i 天时，用户已经连续跑步打卡至少 y_i 天（即第 $x_i - y_i + 1$ 到第 x_i 天均完成了跑步打卡），那么小 T 同学就会请用户吃饭，从而使用户的能量值提高 v_i。

现在大 Y 想知道，在软件试运行的 n 天结束后，他的能量值**最高**可以达到多少？

输入格式

从文件 `run.in` 中读入数据。

本题的测试点包含有多组测试数据。

输入的第一行包含两个整数 c 和 t，分别表示测试点编号和测试数据组数。对于样例，c 表示该样例与测试点 c 拥有相同的限制条件。

接下来，对于每组测试数据：

- 输入的第一行包含四个正整数 n, m, k, d，分别表示试运行的天数、挑战的个数、大 Y 单次跑步打卡的连续天数限制以及大 Y 跑步打卡减少的能量值。
- 接下来 m 行，每行包含三个正整数 x_i, y_i, v_i，表示一次挑战。

输出格式

输出到文件 `run.out` 中。

输出一行一个整数表示对应的答案。

样例 1

Input	Output
1 1 3 2 2 1 2 2 4 3 2 3	2

在第 $1, 2$ 天跑步打卡，第 3 天不跑步打卡，最终会获得 $(-1) + (-1) + 4 = 2$ 的能量值。

样例 2

见附加文件中的 [`run2.in`](file:run2.in) 和 [`run2.ans`](file:run2.ans)。

该组样例满足测试点 3 的限制。

样例 3

见附加文件中的 [`run3.in`](file:run3.in) 和 [`run3.ans`](file:run3.ans)。

该组样例满足测试点 5 的限制。

样例 4

见附加文件中的 [`run4.in`](file:run4.in) 和 [`run4.ans`](file:run4.ans)。

该组样例满足测试点 15 的限制。

样例 5

见附加文件中的 [`run5.in`](file:run5.in) 和 [`run5.ans`](file:run5.ans)。

该组样例满足测试点 17 的限制。

样例 6

见附加文件中的 [`run6.in`](file:run6.in) 和 [`run6.ans`](file:run6.ans)。

该组样例满足测试点 19 的限制。

数据范围与提示

记 $l_i = x_i - y_i + 1$，$r_i = x_i$；

对于所有测试数据，保证：$1 \le t \le 10$，$1 \le k \le n \le 10^9$，$1 \le m \le 10^5$，$1 \le l_i \le r_i \le n$，$1 \le d, v_i \le 10^9$。

测试点编号	$n \le$	$m \le$	特殊性质
$1, 2$	18	10^2	无
$3, 4$	10^2	10^2	无
$5 \sim 7$	10^3	10^3	无
$8, 9$	10^3	10^5	无
$10, 11$	10^5	10^3	无
$12 \sim 14$	10^5	10^5	无
$15, 16$	10^9	10^5	A
$17, 18$	10^9	10^5	B
$19 \sim 21$	10^9	10^5	C
$22 \sim 25$	10^9	10^5	无

特殊性质 A：$k \le 10^2$；

特殊性质 B：$\forall 1 \le i < m$，$r_i < l_{i+1}$；

特殊性质 C：$\forall 1 \le i < j \le m$，$l_i < l_j$，$r_i < r_j$。