

Shift

时间限制：6.0s 内存限制：512M

题目描述

有一个 $n \times m$ 的表格，由 n 行 m 列组成。初始时，每一行都有一个非空字符串 s_i （长度为 $|s_i| \leq m$ ），并且给定该字符串的起始位置 p_i （ $1 \leq p_i \leq m - |s_i| + 1$ ）。也就是说，第 i 个字符串恰好填入表格的第 i 行，占据了第 p_i 到 $p_i + |s_i| - 1$ 列，我们保证这些字符串的放置始终不会超出表格的左右边界。

现在你需要维护 q 次操作，操作分为以下两种：

- `1 x d`：将第 x 行的字符串移动 d 个单位。若 $d < 0$ 表示向左移动 $|d|$ 列，若 $d > 0$ 表示向右移动 d 列。我们保证移动后的字符串占据的区间仍然在 $[1, m]$ 范围内，不会超出表格边界。
- `2 x`：给定 x ，询问当前和第 x 行能够完全对齐且相同字符构成的最长连续子串的长度。具体而言，你需要找出最大的 len （ $len \geq 0$ ），使得存在某一行 y （ $y \neq x$ ）以及某个列首下标 k ，满足第 x 行和第 y 行在横跨第 k 列到第 $k + len - 1$ 列的区域内都有字符分布，且在这段区间内这两行的字符均一一对应相等。如果无法找到任何匹配的字符，则最长连续子串的长度为 0。

输入格式

第一行包含一个整数 T （ $T = 5$ ），表示数据组数。

对于每组测试数据：第一行包含三个整数 n, m, q ，分别表示表格的行数、列数以及操作的次数。接下来 n 行，第 i 行包含一个整数 p_i 和一个字符串 s_i ，表示第 i 行字符串初始时的起始位置和字符串内容。（字符串仅由小写英文字母组成）接下来 q 行，每行描述一个操作。格式为 `1 x d` 或 `2 x`。

- `1 x d` 表示将第 x 行的字符串移动 d 步。
- `2 x` 表示查询第 x 行与其它任意行的最长匹配长度。

对于所有测试数据，保证 $1 \leq n, q \leq 10^4$ ， $1 \leq m \leq 20$ ， $1 \leq |s_i| \leq m$ 。

输出格式

对于每一次 `2 x` 的查询，输出一行包含一个整数，表示在对齐状态下的最长连续相同子串长度。

样例输入

5

3 10 3

1 abcd

2 bcde

5 xyz

2 1

1 2 1

2 1

2 5 2

1 ab

1 ab

2 1

2 2

2 5 2

1 a

2 b

2 1

2 2

2 5 3

1 abc

2 bcd

2 1

1 2 -1

2 1

2 5 2

1 a

2 a

2 1

1 2 -1

样例输出

```
3
0
2
2
0
0
2
0
0
```