Jump the Frog

Input file: `standard input`
Output file: `standard output`
Time limit: 3 seconds
Memory limit: 1024 mebibytes

Have you played "Tap the Frog"? It is a large collection of mini-games where you help a frog with its daily tasks. Today, we will discuss the second mini-game called "Jump the Frog."

The game can be visualized as a long grid strip on a lake, where each cell either contains a lily pad or nothing but the water surface. In the leftmost cell of the strip, there is a lily pad, and a frog is sitting on it. The game interface has k buttons: "jump right by one cell," "jump right by two cells," ..., "jump right by k cells." For simplicity, unlike the original game, we will assume that the frog can only jump on lily pads; also, if there are $r < k$ cells remaining to the right of the frog, you cannot press the button "jump right by i cells" where $i > r$.

You want to create a level for this game. You already have several templates, each of which can be represented as a string of characters "`~`" and "`O`": the character "`~`" with ASCII code 126 represents a cell without a lily pad, and the Latin letter "`O`" with ASCII code 79 represents a cell with a lily pad. Next, you create several new templates: in one operation, you take two templates s_i and s_j (either two different ones or the same one), concatenate them, and add the resulting string $s_i s_j$ to the collection of templates.

For each template, find out how many ways there are to complete the level, that is, to press some buttons in some order to get from the leftmost cell to the rightmost cell without ever visiting cells without a lily pad. Two ways are considered different if they either have a different number of button presses, or they have the same number of presses but for some i, the i-th pressed button in the first way was not the same as in the second. Since these counts can be quite large, find them modulo 998 244 353. Assume that if there is no lily pad in the left or right cell (or in both), the number of ways is zero, even if the strip has a length of one.

Input

The first line contains three integers, n, k, and a: the number of existing templates, the maximum jump length of the frog, and the number of templates you will add ($1 \le n, a \le 10^5$; $1 \le k \le 20$).

In the next n lines, there are strings $s_1, s_2, \ldots, s_n$: the descriptions of the existing templates ($|s_1| + \ldots + |s_n| \le 10^5$). All strings s_i are non-empty and consist of characters "`~`" and "`O`".

The next a lines describe how to construct the templates $n+1, \ldots, n+a$. In the i-th line, there are two integers, ℓ_i and r_i, which mean that the $n+i$-th template is obtained by concatenating the ℓ_i-th and r_i-th templates. Formally, $s_{n+i} = s_{\ell_i} s_{r_i}$ ($1 \le \ell_i, r_i < n+i$).

Output

Output $n+a$ integers $A'_1, A'_2, \ldots, A'_{n+a}$. The number A'_i must be within the range from -2^{63} to $2^{63}-1$ and must be congruent modulo 998 244 353 to the number of ways A_i to get from the left to the right cell of the level, for which the template s_i is chosen (that is, $A_i - A'_i$ must be divisible by 998 244 353).

Example

standard input	*standard output*
4 3 6 0 ~ 000~~ ~000 4 1 1 4 3 1 3 2 8 1 7 7	1 0 0 0 0 3 2 0 0 8