L – Labelled Paths

Time limit: 15 s Memory limit: 512 MiB

We are given a directed acyclic graph with n vertices and m edges. Each edge has a *label* (a string of lowercase letters; possibly even an empty string). We can now extend the concept of labels from edges to paths by defining the *label of a path* as the concatenation of the labels of the edges that constitute this path (in the same order in which they appear in the path). The *smallest path* from a start vertex s to a destination vertex t is the path (from s to t) whose label is lexicographically smallest (i.e. the earliest in lexicographical order) amongst all the paths from s to t. Write a program that, for a given s, outputs the smallest paths from s to t for all vertices t of the graph.

Input data

The first line contains four space-separated integers: n (the number of vertices), m (the number of edges), d (the length of the string A, on which see below) and s (the number of the start vertex). The vertices are numbered by integers from 1 to n.

The second line contains a string A, which is exactly d characters long; all these characters are lowercase letters of the English alphabet. All the edge labels in our graph are substrings of the string A.

The remaining m lines describe the edges of the graph. The i-th of these lines describes the i-th edge and contains four space-separated integers: u_i (the start vertex of this edge), v_i (the end vertex of this edge), p_i and ℓ_i. The last two of these integers indicate that the label of this edge is the substring of A that begins with the p_i-th character of A and is ℓ_i characters long. For this purpose we consider the characters of A to be indexed by integers from 1 to d.

Input limits

- $1 \leq s \leq n \leq 600$
- $1 \leq m \leq 2\,000$
- $1 \leq d \leq 10^6$
- $1 \leq u_i \leq n$, $1 \leq v_i \leq n$, $u_i \neq v_i$ (for all $i = 1, \ldots, m$)
- $1 \leq p_i$, $0 \leq \ell_i$, $p_i + \ell_i - 1 \leq d$ (for all $i = 1, \ldots, m$)
- The graph is acyclic and has no parallel edges (i.e. from $i \neq j$ it follows that $u_i \neq u_j$ and/or $v_i \neq v_j$).

Output data

Output n lines, where the t-th line (for $t = 1, \ldots, n$) describes the smallest path from s to t. If there is no path from s to t, the line should contain only the integer `0` and nothing else. Otherwise the line should start with the number of vertices on the path (including vertices s and t), followed by the list of those vertices, separated by spaces. If there are several possible solutions, you may output any of them.

Example

Input

```
5 7 6 3
abcbca
3 2 1 1
2 1 5 1
2 5 4 2
3 1 1 2
3 4 3 2
1 4 6 1
5 4 5 2
```

Output

```
2 3 1
2 3 2
1 3
3 3 1 4
3 3 2 5
```

Comment

In this example, the edge $3 \to 1$ has the label `ab`; the edge $1 \to 4$ has the label `a`; the smallest path from 3 to 4 is $3 \to 1 \to 4$, whose label is `aba`.