

G. Orientation of Edges

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Vasya has a graph containing both directed (oriented) and undirected (non-oriented) edges. There can be multiple edges between a pair of vertices.

Vasya has picked a vertex s from the graph. Now Vasya wants to create two separate plans:

1. to orient each undirected edge in one of two possible directions to *maximize* number of vertices reachable from vertex s ;
2. to orient each undirected edge in one of two possible directions to *minimize* number of vertices reachable from vertex s .

In each of two plans each undirected edge must become directed. For an edge chosen directions can differ in two plans.

Help Vasya find the plans.

Input

The first line contains three integers n , m and s ($2 \leq n \leq 3 \cdot 10^5$, $1 \leq m \leq 3 \cdot 10^5$, $1 \leq s \leq n$) — number of vertices and edges in the graph, and the vertex Vasya has picked.

The following m lines contain information about the graph edges. Each line contains three integers t_i , u_i and v_i ($1 \leq t_i \leq 2$, $1 \leq u_i, v_i \leq n$, $u_i \neq v_i$) — edge type and vertices connected by the edge. If $t_i = 1$ then the edge is directed and goes from the vertex u_i to the vertex v_i . If $t_i = 2$ then the edge is undirected and it connects the vertices u_i and v_i .

It is guaranteed that there is at least one undirected edge in the graph.

Output

The first two lines should describe the plan which maximizes the number of reachable vertices. The lines three and four should describe the plan which minimizes the number of reachable vertices.

A description of each plan should start with a line containing the number of reachable vertices. The second line of a plan should consist of f symbols '+' and '-', where f is the number of undirected edges in the initial graph. Print '+' as the j -th symbol of the string if the j -th undirected edge (u, v) from the input should be oriented from u to v . Print '-' to signify the opposite direction (from v to u). Consider undirected edges to be numbered in the same order they are given in the input.

If there are multiple solutions, print any of them.

Examples

input
2 2 1 1 1 2 2 2 1
output
2 - 2 +
input

```
6 6 3
2 2 6
1 4 5
2 3 4
1 4 1
1 3 1
2 2 3
```

output

```
6
++-
2
+-+
```