H. Tourist Guide

time limit per test: 2 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

It is not that easy to create a tourist guide as one might expect. A good tourist guide should properly distribute flow of tourists across the country and maximize the revenue stream from the tourism. This is why there is a number of conditions to be met before a guide may be declared as an official tourist guide and approved by Ministry of Tourism.

Ministry of Tourism has created a list of k remarkable cities out of n cities in the country. Basically, it means that in order to conform to strict regulations and to be approved by the ministry a tourist guide should be represented as a set of routes between remarkable cities so that the following conditions are met:

- the first and the last city of every route are distinct remarkable cities,
- each remarkable city can be an endpoint of at most one route,
- there is no pair of routes which share a road.

Please note that a route may pass through a remarkable city. Revenue stream from the tourism highly depends on a number of routes included in a tourist guide so the task is to find out a set of routes conforming the rules of a tourist guide with a maximum number of routes included.

Input

The first line contains three integer numbers $n,\ m,\ k\ (1 \le n \le 50000,\ 0 \le m \le 50000,\ 1 \le k \le n)$ — the number of cities in the country, the number of roads in the country and the number of remarkable cities correspondingly.

Each of the following m lines contains two integer numbers a_i and $b_i\ (1 \le a_i, b_i \le n)$ — meaning that cities a_i and b_i are connected by a bidirectional road. It is guaranteed that a_i and b_i are distinct numbers and there is no more than one road between a pair of cities.

The last line contains k distinct integer numbers — a list of remarkable cities. All cities are numbered from 1 to n.

Output

The first line of the output should contain c — the number of routes in a tourist guide. The following c lines should contain one tourist route each. Every route should be printed in a form of "$t\ v_1\ v_2\ ...\ v_{t+1}$", where t is a number of roads in a route and $v_1,\ v_2, ..., v_{t+1}$ — cities listed in the order they are visited on the route.

If there are multiple answers print any of them.

Examples

input
6 4 4 1 2 2 3 4 5 5 6 1 3 4 6
output
2 2 1 2 3 2 4 5 6

input
4 3 4 1 2 1 3

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

output

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