Problem D: Subway

Johny is going to visit his friend Michelle. His dad allowed him to go there on his own by subway. Johny loves traveling by subway and would gladly use this opportunity to spend half a day underground, but his dad obliged him to make as few line changes as possible. There are a lot of stations in the city, and several subway lines connecting them. All trains are perfectly synchronized – the travel between two consecutive stations on every line takes exactly one minute, and changing lines at any station takes no time at all.

Given the subway map, help Johny to plan his trip so that he can travel for as long as possible, while still following his dad's order.

Input

First line of input contains the number of test cases T. The descriptions of the test cases follow:

The description of each test case starts with an empty line. The next two lines begin with the strings **Stops:** and **Lines:**, and contain the names (separated by a comma and a space) of all subway stops and lines, respectively. A single line for each subway line follows (in no particular order), beginning with <**line-name**> **route:** and listing the names of the stops along this particular line. The final two lines specify the names of the (different) stations nearby Johny's and Michelle's homes.

In each test case, there are at most 300 000 stations and 100 000 lines, whose total length does not exceed 1 000 000. The names of lines and stations are between 1 and 50 characters long and can contain letters, digits, hyphens (**-**), apostrophes (**'**) and "and" signs (**&**). All lines are bidirectional (although changing the direction of travel counts as a line change) and there are no self-crossings.

Output

Print the answers to the test cases in the order in which they appear in the input. For each test case, print a single line summarizing the optimal route Johny can take (see example output for exact format). You may assume that such a route always exists.

Example

Some lines in the example test data below were too long and had to be wrapped. You can access full sample tests at your workstation.

For an example input

```
3

Stops: OxfordCircus, PiccadillyCircus, HydeParkCorner, King'sCross,
GreenPark, Arsenal, Victoria, Highbury&Islington, LeicesterSquare
Lines: Blue, Cyan
Cyan route: Highbury&Islington, King'sCross, OxfordCircus,
GreenPark, Victoria
Blue route: HydeParkCorner, GreenPark, PiccadillyCircus,
LeicesterSquare, King'sCross, Arsenal
Johny lives at King'sCross
Michelle lives at GreenPark

Stops: OxfordCircus, PiccadillyCircus, HydeParkCorner, King'sCross,
GreenPark, Arsenal, Victoria, Highbury&Islington, LeicesterSquare
Lines: Blue, Cyan
Cyan route: Highbury&Islington, King'sCross, OxfordCircus,
GreenPark, Victoria
Blue route: HydeParkCorner, GreenPark, PiccadillyCircus,
LeicesterSquare, King'sCross, Arsenal
Johny lives at PiccadillyCircus
Michelle lives at LeicesterSquare

Stops: OxfordCircus, PiccadillyCircus, HydeParkCorner, King'sCross,
GreenPark, Arsenal, Victoria, Highbury&Islington, LeicesterSquare
Lines: Blue, Cyan
Cyan route: Highbury&Islington, King'sCross, OxfordCircus,
GreenPark, Victoria
Blue route: HydeParkCorner, GreenPark, PiccadillyCircus,
LeicesterSquare, King'sCross, Arsenal
Johny lives at Victoria
Michelle lives at HydeParkCorner
```

the correct answer is:

```
optimal travel from King'sCross to GreenPark: 1 line, 3 minutes
optimal travel from PiccadillyCircus to LeicesterSquare: 1 line, 1 minute
optimal travel from Victoria to HydeParkCorner: 2 lines, 7 minutes
```