# P3

There is a famous railway station in PopPush City. Country there is incredibly hilly. The station was built in last century. Unfortunately, funds were extremely limited that time. It was possible to establish only a surface track. Moreover, it turned out that the station could be only a dead-end one (see picture) and due to lack of available space it could have only one track.
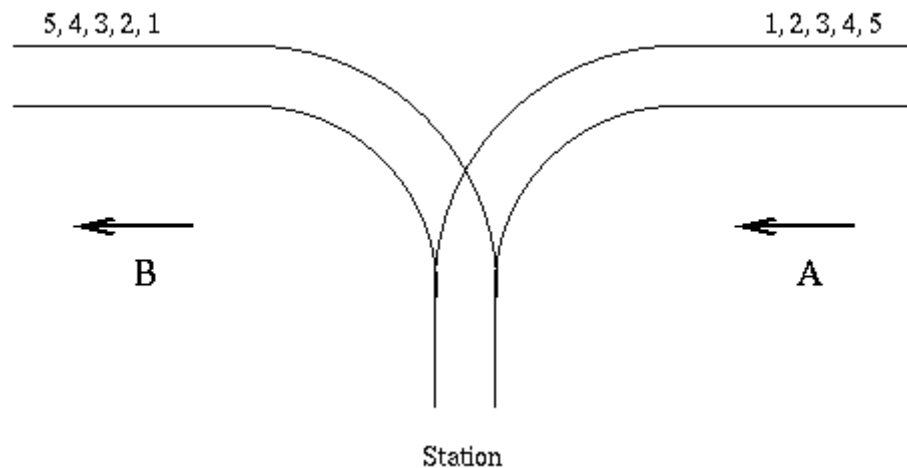
The local tradition is that every train arriving from the direction A continues in the direction B with coaches reorganized in some way. Assume that the train arriving from the direction A has $N \leq 1000$ coaches numbered in increasing order $1, 2, \ldots, N$. The chief for train reorganizations must know whether it is possible to marshal coaches continuing in the direction B so that their order will be $a_1.a_2, \ldots, a_N$. Help him and write a program that decides whether it is possible to get the required order of coaches. You can assume that single coaches can be disconnected from the train before they enter the station and that they can move themselves until they are on the track in the direction B. You can also suppose that at any time there can be located as many coaches as necessary in the station. But once a coach has entered the station it cannot return to the track in the direction A and also once it has left the station in the direction B it cannot return back to the station.

Input

The input file consists of blocks of lines. Each block except the last describes one train and possibly more requirements for its reorganization. In the first line of the block there is the integer N described above. In each of the next lines of the block there is a permutation of $1, 2, \ldots, N$. The last line of the block contains just '0'.

The last block consists of just one line containing '0'.

Output

The output file contains the lines corresponding to the lines with permutations in the input file. A line of the output file contains 'Yes' if it is possible to marshal the coaches in the order required on the corresponding line of the input file. Otherwise it contains 'No'. In addition, there is one empty line after the lines corresponding to one block of the input file. There is no line in the output file corresponding to the last "null" block of the input file.

Sample Input

```
5
1 2 3 4 5
5 4 1 2 3
0
6
6 5 4 3 2 1
0
0
```

Sample Output

```
Yes
No

Yes
```

P4

In a small city called Iokh, a train service, **Airport-Express**, takes residents to the airport more quickly than other transports. There are two types of trains in Airport-Express, the **Economy-Xpress** and the **Commercial-Xpress**. They travel at different speeds, take different routes and have different costs.

Jason is going to the airport to meet his friend. He wants to take the Commercial-Xpress which is supposed to be faster, but he doesn't have enough money. Luckily he has a ticket for the Commercial-Xpress which can take him one station forward. If he used the ticket wisely, he might end up saving a lot of time. However, choosing the best time to use the ticket is not easy for him.

Jason now seeks your help. The routes of the two types of trains are given. Please write a program to find the best route to the destination. The program should also tell when the ticket should be used.

Input

The input consists of several test cases. Consecutive cases are separated by a blank line.

The first line of each case contains 3 integers, namely N, S and E ($2 \leq N \leq 500, 1 \leq S, E \leq N$), which represent the number of stations, the starting point and where the airport is located respectively.

There is an integer M ($1 \leq M \leq 1000$) representing the number of connections between the stations of the Economy-Xpress. The next M lines give the information of the routes of the Economy-Xpress. Each consists of three integers X, Y and Z ($X, Y \leq N, 1 \leq Z \leq 100$). This means X and Y are connected and it takes Z minutes to travel between these two stations.

The next line is another integer K ($1 \leq K \leq 1000$) representing the number of connections between the stations of the Commercial-Xpress. The next K lines contain the information of the Commercial-Xpress in the same format as that of the Economy-Xpress.

All connections are bi-directional. You may assume that there is exactly one optimal route to the airport. There might be cases where you MUST use your ticket in order to reach the airport.

Output

For each case, you should first list the number of stations which Jason would visit in order. On the next line, output '`Ticket Not Used`' if you decided NOT to use the ticket; otherwise, state the station where Jason should get on the train of Commercial-Xpress. Finally, print the total time for the journey on the last line. Consecutive sets of output must be separated by a blank line.

Sample Input

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

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

Sample Output

```
1 2 4
2
5
```