

Free DIY Tour

<https://vjudge.net/problem/hdu-1224>

Weiwei is a software engineer of ShiningSoft. He has just excellently fulfilled a software project with his fellow workers. His boss is so satisfied with their job that he decide to provide them a free tour around the world. It's a good chance to relax themselves. To most of them, it's the first time to go abroad so they decide to make a collective tour.

The tour company shows them a new kind of tour circuit - DIY circuit. Each circuit contains some cities which can be selected by tourists themselves. According to the company's statistic, each city has its own interesting point. For instance, Paris has its interesting point of 90, New York has its interesting point of 70, ect. Not any two cities in the world have straight flight so the tour company provide a map to tell its tourists whether they can got a straight flight between any two cities on the map. In order to fly back, the company has made it impossible to make a circle-flight on the half way, using the cities on the map. That is, they marked each city on the map with one number, a city with higher number has no straight flight to a city with lower number.

Note: Weiwei always starts from Hangzhou(in this problem, we assume Hangzhou is always the first city and also the last city, so we mark Hangzhou both 1 and N+1), and its interesting point is always 0.

Now as the leader of the team, Weiwei wants to make a tour as interesting as possible. If you were Weiwei, how did you DIY it?

Input

The input will contain several cases. The first line is an integer T which suggests the number of cases. Then T cases follows. Each case will begin with an integer N( $2 \leq N \leq 100$ ) which is the number of cities on the map. Then N integers follows, representing the interesting point list of the cities. And then it is an integer M followed by M pairs of integers  $[A_i, B_i]$  ( $1 \leq i \leq M$ ). Each pair of  $[A_i, B_i]$  indicates that a straight flight is available from City  $A_i$  to City  $B_i$ .

Output

For each case, your task is to output the maximal summation of interesting points Weiwei and his fellow workers can get through optimal DIYing and the optimal circuit. The format is as the sample. You may assume that there is only one optimal circuit.

Output a blank line between two cases.

Sample

Input	Output
2 3 0 70 90 4 1 2 1 3 2 4 3 4 3 0 90 70 4 1 2 1 3 2 4 3 4	CASE 1# points : 90 circuit : 1->3->1  CASE 2# points : 90 circuit : 1->2->1