



Extended Traffic

Link: http://www.lightoj.com/volume_showproblem.php?problem=1074

Solution:

C++	http://ideone.com/AGs6Y0
Java	https://ideone.com/WiP5r3
Python	https://ideone.com/mWEtWz

*** Lưu ý: nếu bạn lập trình bằng ngôn ngữ python bạn submit trên trang web www.algote.com

Tóm tắt đề:

Bài toán cho bạn danh sách các đỉnh n (điểm giao thông) và trọng số tại mỗi điểm này. Cho bạn m đường đi giao thông khác nhau. Mỗi đường này có trọng số là (điểm đến – điểm đi)³

Sau đó đề cho bạn q (truy vấn) là danh sách các đỉnh cần tìm đường đi ngắn nhất từ đỉnh 1 đến các đỉnh này. Nhiệm vụ của bạn là tìm đường đi ngắn nhất.

Input

Dòng đầu tiên chứa số T ($T \leq 50$). Là số lượng bộ test. Mỗi bộ test chứa các thông tin như sau:

Dòng đầu chứa số n là số lượng các cụm giao thông ($1 < n \leq 200$).

Dòng tiếp theo lần lượt là n số, đại diện cho sự “bận rộn” của nút giao thông từ $1 \rightarrow n$. Các số này là số nguyên dương ≤ 20 .

Dòng tiếp theo là số m là số lượng đường trong thành phố.

m dòng tiếp theo mỗi dòng chứa 2 số là đường đi từ đỉnh nguồn đến đỉnh đích.

Dòng tiếp theo chứa số q là số lượng truy vấn, q dòng tiếp theo mỗi dòng chứa một số là đỉnh mà bạn cần tìm đường đi ngắn nhất từ đỉnh 1 đến đỉnh đó.

Output

Với mỗi truy vấn bạn sẽ in ra được đi ngắn nhất từ đỉnh 1 đến đỉnh đó. Nếu chi phí nhỏ hơn 3 thì bạn in ra “?”.

2	Case 1: 3 4
---	-------------------

5	Case 2:
6 7 8 9 10	?
6	
1 2	
2 3	
3 4	
1 5	
5 4	
4 5	
2	
4	
5	
2	
10 10	
1	
1 2	
1	
2	

Giải thích:

Có 2 bộ testcase. Bộ thứ nhất gồm các thông tin:

Có 5 điểm giao thông, lần lượt có các trọng số là 6, 7, 8, 9, 10.

Có 6 đường đi, lần lượt là (1, 2), (2, 3), (3, 4), (1, 5), (5, 4), (4, 5).

Có 2 truy vấn cần tìm đường đi từ đỉnh 1 đến đỉnh 4 và từ đỉnh 1 đến đỉnh 5.

Tương tự bộ testcase thứ 2 bạn có thể suy luận như testcase số 1.

Hướng dẫn giải: Đề không cho trọng số mỗi cạnh. Bạn phải dùng trọng số tại mỗi điểm để tìm trọng số cho mỗi cạnh bằng công thức (điểm đến – điểm đi)³. Sau khi tìm trọng số xong dùng Bellman-Ford để tìm đường đi ngắn nhất từ đỉnh 1 đến các truy vấn.

Độ phức tạp: $O(T * V * E)$ với V là số đỉnh (số điểm giao thông), E là số đường đi, T là số bộ test.