COCI '13 Contest 2 #4 Putnik

Time Limit: 0.6s **Memory Limit:** 32M
Python: 128M

Chances are that you have probably already heard of the travelling salesman problem. If you have, then you are aware that it is an NP-hard problem because it lacks an efficient solution. Well, this task is an uncommon version of the famous problem! Its uncommonness derives from the fact that this version is, actually, solvable.

The travelling salesman is on a mission to visit N cities, each exactly once. The cities are represented by numbers $1, 2, \ldots, N$. What we know is the direct flight duration between each pair of cities. The salesman, being the efficient man that he is, wants to modify the city visiting sequence so that the total flight duration is the minimum possible.

Alas, all is not so simple. In addition, the salesman has a peculiar condition regarding the sequence. For **each** city labeled K must apply: either all cities with **labels smaller than** K have been visited before the city labeled K or they will all be visited after the city labeled K. In other words, the situation when one of such cities is visited before, and the other after is not allowed.

Assist the poor fellow in his ambitious mission and calculate the minimum total flight duration needed in order to travel to all the cities, starting from whichever and ending in whichever city, visiting every city exactly once, so that his peculiar request is fulfilled.

Input

The first line of input contains the positive integer N $(2 \leq N \leq 1\,500)$, the number of cities.

Each of the following N lines contains N positive integers from the interval $[0, 1000]$. The number in B^{th} place in the A^{th} row represents the flight duration between cities A and B; that number is equal to the A^{th} number in the B^{th} row. When $A = B$, that number is 0. Otherwise, it is a positive value.

Output

The first and only line of output must contain the required minimum total flight duration.

Scoring

In test data worth 1/3 of total points, N will be at most 10.

In test data worth 1/2 of total points, N will be at most 20.

Sample Input 1

```
3
0 5 2
5 0 4
2 4 0
```

Sample Output 1

```
7
```

Explanation for Sample Output 1

The optimal sequence is 2, 1, 3 or 3, 1, 2. The sequence 1, 3, 2 is even more favourable, but it does not fulfill the salesman's condition.

Sample Input 2

```
4
0 15 7 8
15 0 16 9
7 16 0 12
8 9 12 0
```

Sample Output 2

```
31
```

Explanation for Sample Output 2

The sequence is either 3, 1, 2, 4 or 4, 2, 1, 3.

COCI '20 Contest 6 #2 Alias

Time Limit: 1.0s **Memory Limit:** 512M

Novak and Rafael are playing a simplified version of the game Alias. Novak needs to make Rafael guess a word without saying it. Rafael has a database of n words in his head, and there are m connections between some words. The connection between words x and y, with time t, means that if Rafael remembers the word x or hears it, after t milliseconds he will remember the word y.

Novak and Rafael will play q rounds. In each round, Novak wants to know: if he says the word a, after how many milliseconds will Rafael remember the word b for the first time? The rounds are independent.

Input Specification

The first line contains integers n $(2 \leq n \leq 1\,000)$ and m $(1 \leq m \leq 1\,000)$, the number of words and the number of connections.

Each of the following m lines contains two different words x_i and y_i, and an integer t_i $(1 \leq t_i \leq 10^9)$, that describe a connection. The words consist of at most 20 lowercase letters. All words from Rafael's database will appear at least once. It is possible that there are multiple connections between some pairs of words.

The following line contains an integer q $(1 \leq q \leq 1\,000)$, the number of rounds.

Each of the following q lines contains two different words a_i and b_i, the word that Novak will say and the word that Rafael needs to remember in the i^{th} round. Both words appear in Rafael's database.

Output Specification

Output q lines. In the i^{th} line output the time for the i^{th} round in milliseconds, or `Roger` if Rafael will never remember the word.

Constraints

Subtask	Points	Constraints
1	20	$2 \leq n \leq 10$
2	20	$2 \leq n \leq 100$
3	30	No additional constraints.

Sample Input 1

```
3 2
novak goat 1
goat simulator 3
2
novak simulator
simulator goat
```

Sample Output 1

```
4
Roger
```

Explanation for Sample Output 1

In the first round, Novak will say the word `novak`. After 1 millisecond, Rafael will remember the word `goat`, and after 3 more milliseconds the required word `simulator`. In the second round, Novak will say the word `simulator`, but Rafael won't remember any other words.

Sample Input 2

```
3 3
kile legend 4
legend beer 5
beer kile 6
2
kile beer
legend kile
```

Sample Output 2

```
9
11
```

Sample Input 3

```
4 5
rafael me 5
me ow 6
ow ausopenfinal 2012
ausopenfinal me 2
rafael ausopenfinal 2
3
rafael me
me rafael
ow me
```

Sample Output 3

```
4
Roger
2014
```