# Travel Optimizer

*The newest startup is called Travel Optimizer. The problem that the company solves is to find the shortest path from one city to another with at most k stops. You are hired to write the main functionality of the app.*

We have a set of towns and some of them are connected by bi-direct paths. Each path has a price that has to be paid.

Write a program that prints the **cheapest path from a starting city to the destination city with at most k stops**.

## Input

* On the first, you will receive an integer – e – number of edges.
* On the next e lines, you will receive an edge in the following format: "{start}, {end}, {weight}".
* On the next line, you will receive a start city.
* On the last line, you will receive a destination city.
* On the next line, you will receive an integer – k – how many stops you can use.

## Output

* Print all cities that form the cheapest path separated by a space.
* If the destination city can’t be reached from the starting point, then you have to print: "There is no such path."

## Constraints

* e will be in the range [1… 50].
* Cities will always be represented by positive integers.
* Prices will always be positive integers.
* Start city and destination city will always be valid cities.
* k will be in the range [1… 10].

## Examples

|  |  |  |
| --- | --- | --- |
| **Input** | **Output** | **Comments** |
| 3  0, 1, 15  1, 2, 5  0, 2, 50  0  2  0 | 0 2 | * Here, the cheapest path is 0 -> 1 -> 2 (20), however, we have to make a stop in city with id 1, but we have k = 0, so we can’t make any stops. |
| 5  0, 1, 1  1, 2, 5  0, 2, 50  1, 3, 25  3, 2, 10  0  3  3 | 0 1 2 3 |  |