apac test
powered by Google Code Jam

Time Remaining: 2 hours 10 min Rank: 183 Score: 0

**lanfeng19911015@gmail.com** |
Contest scoreboard | Sign out

Round B APAC Test 2016

**A. Travel**

B. gWheels

C. gNumbers

D. Albocede DNA

Ask a question

View my submissions

- Submissions

Travel

| | |
|---|---|
| 6pt | Not attempted<br>**78/401 users** correct (19%) |
| 12pt | Not attempted<br>**70 users** attempted |

gWheels

| | |
|---|---|
| 5pt | Not attempted<br>**101/245 users** correct (41%) |
| 14pt | Not attempted<br>**39 users** attempted |

gNumbers

| | |
|---|---|
| 8pt | Not attempted<br>**8/96 users** correct (8%) |
| 16pt | Not attempted<br>**4 users** attempted |

Albocede DNA

| | |
|---|---|
| 16pt | Not attempted<br>**0/29 users** correct (0%) |
| 23pt | Not attempted |

- Top Scores

| | |
|---|---|
| kcm1700 | 61 |
| imamur | 37 |
| abcsampson | 37 |
| yaray | 37 |
| tapasjain01 | 37 |
| himanshujaju | 24 |
| Mr.Fury | 24 |
| mkrjn99 | 24 |
| johngs | 23 |
| KKOrange | 19 |

# Problem A. Travel

Confused? Read the quick-start guide.

| | |
|---|---|
| Small input<br>6 points | Solve A-small<br>You may try multiple times, with penalties for wrong submissions. |
| Large input<br>12 points | You must solve the small input first.<br>You have 8 minutes to solve 1 input file. (Judged after contest.) |

## Problem

There are **N** cities in Chelsea's state (numbered starting from 1, which is Chelsea's city), and **M** bidirectional roads directly connect them. (A pair of cities may even be directly connected by more than one road.) Because of changes in traffic patterns, it may take different amounts of time to use a road at different times of day, depending on when the journey starts. (However, the direction traveled on the road does not matter – traffic is always equally bad in both directions!) All trips on a road start (and end) exactly on the hour, and a trip on one road can be started instantaneously after finishing a trip on another road.

Chelsea loves to travel and is deciding where to go for her winter holiday trip. She wonders how quickly she can get from her city to various other destination cities, depending on what time she leaves her city. (Her route to her destination may include other intermediate cities on the way.) Can you answer all of her questions?

## Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow.

The first line of each test case contains three integers: the number **N** of cities, the number **M** of roads, and the number **K** of Chelsea's questions.

2**M** lines -- **M** pairs of two lines -- follow. In each pair, the first line contains two different integers x and y that describe one bidirectional road between the x-th city and the y-th city. The second line contains 24 integers Cost[t] ($0 \le t \le 23$) that indicate the time cost, in hours, to use the road when departing at t o'clock on that road. It is guaranteed that $Cost[t] \le Cost[t+1]+1$ ($0 \le t \le 22$) and $Cost[23] \le Cost[0]+1$.

Then, an additional **K** lines follow. Each contains two integers **D** and **S** that comprise a question: what is the fewest number of hours it will take to get from city 1 to city **D**, if Chelsea departs city 1 at **S** o'clock?

## Output

For each test case, output one line containing "Case #x: ", where x is the case number (starting from 1), followed by **K** distinct space-separated integers that are the answers to the questions, in order. If Chelsea cannot reach the destination city for a question, no matter which roads she takes, then output -1 for that question.

## Limits

$1 \le x, y \le$ **N**.
1 ≤ all Cost values ≤ 50.
$1 \le$ **D** $\le$ **N**.
$0 \le$ **S** $\le 23$.

## Small dataset

$1 \le$ **T** $\le 100$.
$2 \le$ **N** $\le 20$.
$1 \le$ **M** $\le 100$.

$1 \le \mathbf{K} \le 100$.

### Large dataset

$1 \le \mathbf{T} \le 5$.
$2 \le \mathbf{N} \le 500$.
$1 \le \mathbf{M} \le 2000$.
$1 \le \mathbf{K} \le 5000$.

### Sample

#### Input

```
3
3 3 2
1 2
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 3
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
2 3
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 1
3 3
3 1 2
1 2
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2
3 4
3 3 3
1 2
7 23 23 25 26 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8
1 3
10 11 15 26 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 1
2 3
7 29 28 27 26 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8
2 14
3 3
3 21
```

#### Output

```
Case #1: 1 2
Case #2: 1 -1
Case #3: 17 26 13
```