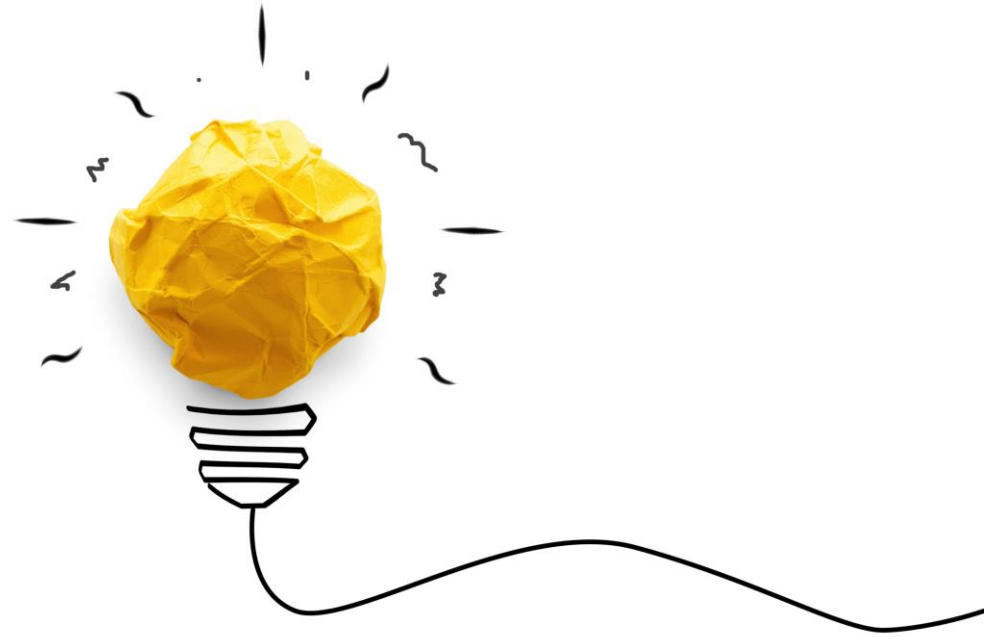


창의적 문제 해결

실습과제 8

공주대학교 컴퓨터공학과

김영부



- 최단경로

<문제 1> 플로이드

Q

37

플로이드

1회 2회 3회



난이도 ●○○ | 풀이 시간 40분 | 시간 제한 1초 | 메모리 제한 256MB | 기출 핵심 유형

링크 <https://www.acmicpc.net/problem/11404>

n ($1 \leq n \leq 100$)개의 도시가 있고, 한 도시에서 출발하여 다른 도시에 도착하는 m ($1 \leq m \leq 100,000$)개의 버스가 있습니다. 각 버스는 한 번 사용할 때 필요한 비용이 있습니다. 모든 도시의 쌍 (A, B)에 대해서 도시 A 에서 B 로 가는 데 필요한 비용의 최솟값을 구하는 프로그램을 작성하세요.

입력 조건

- 첫째 줄에 도시의 개수 n ($1 \leq n \leq 100$)이 주어집니다.
- 둘째 줄에는 버스의 개수 m ($1 \leq m \leq 100,000$)이 주어집니다.
- 셋째 줄부터 $m + 2$ 줄까지 다음과 같은 버스의 정보가 주어집니다. 먼저 처음에는 그 버스의 출발 도시의 번호가 주어집니다. 버스의 정보는 버스의 시작 도시 a , 도착 도시 b , 한 번 타는 데 필요한 비용 c 로 이루어져 있습니다. 시작 도시와 도착 도시가 같은 경우는 없습니다. 비용은 100,000보다 작거나 같은 자연수입니다.
- 시작 도시와 도착 도시를 연결하는 노선은 하나가 아닐 수 있습니다.

출력 조건

- n 개의 줄을 출력해야 합니다. i 번째 줄에 출력하는 j 번째 숫자는 도시 i 에서 j 로 가는 데 필요한 최소 비용입니다. 만약, i 에서 j 로 갈 수 없는 경우에는 그 자리에 0을 출력합니다.

<문제 1> 플로이드

출력 조건 • n개의 줄을 출력해야 합니다. i번째 줄에 출력하는 j번째 숫자는 도시 i에서 j로 가는 데 필요한 최소 비용입니다. 만약, i에서 j로 갈 수 없는 경우에는 그 자리에 0을 출력합니다.

입력 예시

```
5
14
1 2 2
1 3 3
1 4 1
1 5 10
2 4 2
3 4 1
3 5 1
4 5 3
3 5 10
3 1 8
1 4 2
5 1 7
3 4 2
5 2 4
```

출력 예시

```
0 2 3 1 4
12 0 15 2 5
8 5 0 1 1
10 7 13 0 3
7 4 10 6 0
```

<문제 2> 화성 탐사

Q

39

화성 탐사

1회 2회 3회
☐ ☐ ☐

난이도 ●●○ | 풀이 시간 40분 | 시간 제한 1초 | 메모리 제한 128MB | 기출 ACM-ICPC

당신은 화성 탐사 기계를 개발하는 프로그래머입니다. 그런데 화성은 에너지 공급원을 찾기가 어렵습니다. 그래서 에너지를 효율적으로 사용하고자 화성 탐사 기계가 출발 지점에서 목표 지점까지 이동할 때 항상 최적의 경로를 찾도록 개발해야 합니다.

화성 탐사 기계가 존재하는 공간은 $N \times N$ 크기의 2차원 공간이며, 각각의 칸을 지나기 위한 비용(에너지 소모량)이 존재합니다. 가장 왼쪽 위 칸인 $[0][0]$ 위치에서 가장 오른쪽 아래 칸인 $[N - 1][N - 1]$ 위치로 이동하는 최소 비용을 출력하는 프로그램을 작성하세요. 화성 탐사 기계는 특정한 위치에서 상하좌우 인접한 곳으로 1칸씩 이동할 수 있습니다.

입력 조건

- 첫째 줄에 테스트 케이스의 수 T ($1 \leq T \leq 10$)가 주어집니다.
- 매 테스트 케이스의 첫째 줄에는 탐사 공간의 크기를 의미하는 정수 N 이 주어집니다. ($2 \leq N \leq 125$)
이어서 N 개의 줄에 걸쳐 각 칸의 비용이 주어지며 공백으로 구분합니다. ($0 \leq$ 각 칸의 비용 ≤ 9)

출력 조건

각 테스트 케이스마다 $[0][0]$ 의 위치에서 $[N - 1][N - 1]$ 의 위치로 이동하는 최소 비용을 한 줄에 하나씩 출력합니다.

<문제 2> 화성 탐사

입력 예시

```
3
3
5 5 4
3 9 1
3 2 7
5
3 7 2 0 1
2 8 0 9 1
1 2 1 8 1
9 8 9 2 0
3 6 5 1 5
7
9 0 5 1 1 5 3
4 1 2 1 6 5 3
0 7 6 1 6 8 5
1 1 7 8 3 2 3
9 4 0 7 6 4 1
5 8 3 2 4 8 3
7 4 8 4 8 3 4
```

출력 예시

```
20
19
36
```