

최소 비행료

현수는 여름 휴가를 가려고 합니다. 현수가 여름휴가를 계획한 곳은 0번부터 $n-1$ 번으로 구분하는 n 개의 도시가 있습니다. 현수는 비행기로 목적지 도시까지 가려고 합니다.

현수에게는 각 비행기의 운항정보가 있습니다. 어떤 비행기의 운항정보가 $[1, 2, 5]$ 라면 1번 도시에서 2번 도시로 가는 비행기이며 비행료는 5라는 의미입니다. 모든 비행기는 출발도시에서 도착도시로 직항만 합니다. 현수가 사는 도시에서 목적지 도시까지 비행기 직항이 없으면 환승을 해서 목적지 도시까지 가야 합니다.

매개변수 n 에 도시의 개수, 매개변수 $flights$ 에 각 비행기의 운항정보, s 에 현수가 사는 도시, e 에 현수가 여행을 가는 목적지 도시가 주어지고, 매개변수 k 에 값이 주어지면 현수가 최대 k 번 환승을 해서 목적지 도시까지 가는데 드는 최소비용을 반환하는 프로그램을 작성하세요.

현수가 목적지 도시까지 갈 수 없으면 -1 를 반환합니다.

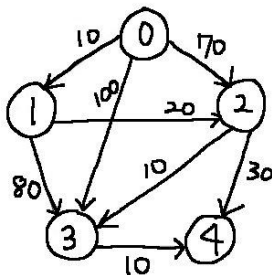
입출력 예:

n	flights	s	e	k	answer
5	[[0, 1, 10], [1, 2, 20], [0, 2, 70], [0, 3, 100], [1, 3, 80], [2, 3, 10], [2, 4, 30], [3, 4, 10]]	0	3	1	80
4	[[0, 1, 10], [0, 2, 10], [1, 3, 5], [2, 3, 3]]	0	3	0	-1
8	[[0, 3, 10], [1, 5, 10], [1, 7, 100], [0, 1, 10], [0, 2, 10], [5, 7, 30], [3, 7, 10], [1, 3, 5], [2, 3, 3]]	1	7	2	15
10	[[1, 8, 50], [0, 8, 30], [1, 0, 10], [2, 8, 10], [0, 3, 10], [1, 5, 10], [1, 7, 100], [0, 1, 10], [0, 2, 10], [5, 7, 30], [3, 7, 10], [1, 3, 5], [2, 3, 3]]	1	8	2	30

제한사항:

- $2 \leq n, s, e \leq 100, 0 \leq k \leq 20$
- flight의 길이는 $n*(n-1) / 2$ 을 넘지 않습니다.
- $flight[i][2]$ 인 비행료는 10,000을 넘지 않습니다.
- $s \neq e$

입력예제 1번 설명 :



• 0번 → 2번 → 3번과 같이 이동하면 2번 도시에서 한 번 환승하고, $70 + 10 = 80$ 이 최소 비용입니다.

• 0번 → 1번 → 2번 → 3번과 같이 이동하면 비용은 $10 + 20 + 10 = 40$ 으로 비용이 작지만 1번과 2번 도시에서 환승을 해야 하므로 답이 될 수 없습니다.

최소 환승 경로

지하철 노선 정보가 담겨있는 routes배열이 주어집니다. routes[i]는 i번 지하철이 운행하는 노선의 역번호가 담겨있습니다.

만약 routes[i]=[1, 2, 7]이면 i번 지하철은 1-->2-->7-->2-->1-->2-->7처럼 역을 반복해서 운행하는 순환지하철입니다.

매개변수 routes에 지하철 노선정보가 주어지고, 매개변수 s에 출발역 번호, 매개변수 e에 도착역 번호가 주어지면 출발지에서 도착지까지의 최소 환승 경로로 이동했을 때

최소환승횟수를 반환하는 프로그램을 작성하세요.

출발역에서 도착역까지 갈 수 없다면 -1을 반환합니다.

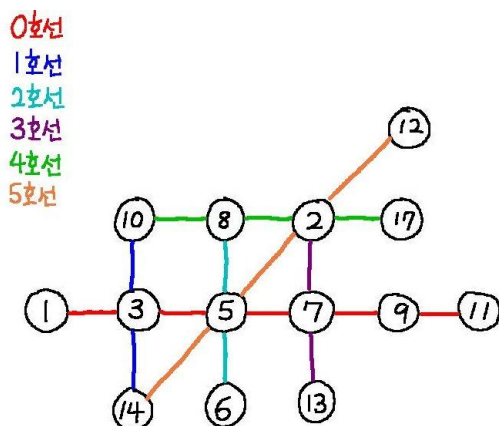
입출력 예:

routes	s	e	answer
[[1, 3, 5, 7, 9, 11], [10, 3, 14], [8, 5, 6], [2, 7, 13], [10, 8, 2, 17], [14, 5, 2, 12]]	1	12	1
[[1, 3, 5, 7], [9, 3, 12], [6, 5, 8], [2, 8, 14, 15], [2, 14, 16]]	1	14	2
[[7, 12],[5, 19],[7, 19],[9, 12, 13],[9, 5, 15]]	9	19	1
[[1, 2, 3, 4, 5, 6, 7],[3, 6, 7]]	1	6	0
[[1, 2, 3, 4, 5],[9, 7, 10],[7, 6, 3, 8], [5, 11, 8, 12]]	1	10	2

제한사항:

- routes의 길이는 1부터 1,000까지입니다. 각 노선은 2개 이상의 역으로 구성됩니다.
- 역번호는 0번부터 1,000,000까지입니다.
- 출발역과 도착역은 같지 않습니다.

입력예제 5번 설명 :



0호선의 1번 역에서 출발해서 5번 역에서 5호선으로 환승해 12번 역으로 가면 됩니다.

벽 허물기

현수에게 $n * m$ 크기의 격자판으로 된 지도정보가 주어집니다. 현수는 격자로 표현된 지도의 0행 0열(격자의 왼쪽 가장 위) 지점에서 $n-1$ 행 $m-1$ 열(격자의 오른쪽 가장 아래 지점)으로 이동하려고 합니다.

지도에서 0값은 통로이고, 1은 벽입니다.

매개변수 board에 지도정보가 주어지면 현수가 (0, 0) 지점에서 (n-1, m-1)지점까지 가기 위해서 허물어야 하는 최소 벽의 개수를 반환하는 프로그램을 작성하세요.

입출력 예:

board	answer
[[0, 1, 1, 0], [1, 0, 1, 1], [0, 1, 0, 0]]	2
[[0, 1, 1, 0], [1, 1, 0, 1], [0, 0, 1, 0], [0, 1, 1, 1], [0, 1, 1, 0]]	3
[[0, 1, 1, 0, 1, 1], [0, 1, 1, 1, 1, 1], [1, 0, 0, 0, 1, 1], [1, 1, 0, 1, 1, 1], [1, 1, 0, 1, 1, 0], [1, 0, 0, 1, 1, 1], [1, 1, 1, 1, 1, 0]]	4
[[0, 1, 1, 0, 1, 1, 1], [1, 1, 1, 0, 1, 1, 1], [1, 0, 0, 0, 0, 1, 1], [1, 1, 1, 0, 1, 1, 1], [1, 1, 1, 0, 1, 1, 0], [1, 0, 1, 0, 1, 1, 1], [1, 1, 1, 1, 1, 1, 1], [1, 1, 1, 1, 1, 1, 1], [1, 1, 1, 1, 1, 1, 0]]	7
[[0, 0, 1, 0, 1, 1, 1], [1, 1, 0, 0, 1, 1, 1], [1, 1, 0, 1, 0, 1, 1], [0, 0, 1, 0, 1, 1, 1], [1, 0, 1, 0, 1, 1, 0], [1, 0, 1, 0, 1, 1, 1], [1, 0, 0, 1, 1, 1, 1], [1, 1, 0, 0, 1, 1, 1], [1, 1, 0, 1, 1, 1, 0]]	5

제한사항:

- $3 \leq n, m \leq 100$
- (0, 0)과 (n-1, m-1) 지점은 벽이 아닙니다.

입력예제 1번 설명 :

0	1	1	0
1	0	1	1
0	1	0	0

현수의 이동 경로는 (0, 0) - (0, 1) - (1, 1) - (1, 2) - (2, 2) - (2, 3) 과 같이 이동하면 벽을 부순 지점은 (0, 1)과 (1, 2)로 현수는 최소 2개의 벽을 허물어야 (2, 3)에 도착할 수 있습니다.

방향 바꾸기

현수에게 $n \times m$ 크기의 격자판으로 된 지도정보가 주어집니다.

지도의 각 격자에는 1, 2, 3, 4의 값이 있는데

1 : 오른쪽의 인접한 격자로 이동을 의미합니다.

2 : 왼쪽의 인접한 격자로 이동을 의미합니다.

3 : 아래로 인접한 격자로 이동을 의미합니다.

4 : 위로 인접한 격자로 이동을 의미합니다.

현수는 격자로 표현된 방향지시대로 0행 0열(격자의 왼쪽 가장 위) 지점에서 $n-1$ 행 $m-1$ 열(격자의 오른쪽 가장 아래 지점)으로 이동하려고 합니다.

매개변수 board에 지도정보가 주어지면 현수가 (0, 0) 지점에서 $(n-1, m-1)$ 지점까지 가기 위해서 방향을 바꾸어야 하는 최소 격자의 개수를 반환하는 프로그램을 작성하세요.

각 격자의 방향은 현수가 원하는 방향으로 오직 한 번만 바꿀 수 있습니다.

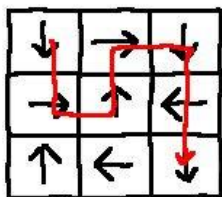
입출력 예:

board	answer
[[3, 1, 3], [1, 4, 2], [4, 2, 3]]	1
[[3, 2, 1, 3], [1, 1, 4, 2], [3, 4, 2, 1], [1, 2, 2, 4]]	2
[[3, 2, 1, 3, 1, 2], [2, 1, 1, 1, 4, 2], [2, 2, 2, 1, 2, 2], [1, 3, 3, 4, 4, 4], [1, 2, 2, 3, 3, 4]]	5
[[3, 2, 1, 3, 1, 2, 2, 2], [2, 1, 1, 1, 4, 2, 1, 1], [2, 2, 2, 1, 2, 2, 3, 4], [1, 3, 3, 4, 4, 4, 3, 1], [1, 2, 2, 3, 3, 4, 3, 4], [1, 2, 2, 3, 3, 1, 1, 1]]	4
[[1, 2, 3, 2, 1, 3, 1, 2, 2, 2], [1, 2, 2, 1, 1, 1, 4, 2, 1, 1], [3, 2, 2, 2, 2, 1, 2, 2, 3, 4], [3, 3, 1, 3, 3, 4, 4, 4, 3, 1], [1, 1, 1, 2, 2, 3, 3, 4, 3, 4], [1, 1, 1, 2, 2, 3, 3, 1, 1, 1]]	5

제한사항:

- $3 \leq n, m \leq 100$
- (0, 0)과 $(n-1, m-1)$ 에도 방향지시는 있습니다.

입력예제 1번 설명 :



현수의 이동 경로는 (0, 0) - (1, 0) - (1, 1) - (0, 1) - (0, 2) - (1, 2) - (2, 2) 과 같이 이동하면 (1, 2) 지점에서만 방향을 바꾸면 됩니다.

이사하기

현수는 고객에게 최소비용으로 이사할 수 있는 이삿짐 센터를 찾아주는 앱을 개발하려고 합니다. 현수가 사는 나라는 각 도시가 양방향으로 이동하는 도로로 연결되어 있고, 모든 도시에는 이삿짐 센터가 하나씩 존재합니다.

도시에 하나씩 존재하는 이삿짐 센터들은 이삿짐 차량을 1대 이상 가지고 있습니다.

각 차량은 운반할 수 있는 이삿짐량의 최대무게, 1km당 이사비용이 정해져 있습니다.

모든 차량은 운반할 수 있는 이삿짐량의 최대무게가 클수록 1km당 이사비용이 더 많아집니다.

이사 비용은 (차량이 이동한 거리(km)) * (해당 차량의 1km당 이용 요금)으로 책정됩니다.

차량이 이동한 거리는 이사 업체가 위치한 도시에서 고객이 있는 출발 도시, 출발 도시에서 도착 도시까지 최단 거리로 이동한 거리입니다.

매개변수 cities에 도시의 이름이 주어지고, 매개변수 edges에 도시를 연결하는 도로정보, 매개변수 cars에 각 이삿짐 센터들이 가지고 있는 차량정보, customers에 고객이 이사정보가 주어지면 고객이 최소비용으로 이사할 수 있는 업체가 존재하는 도시의 이름을 반환하는 프로그램을 작성하세요.

만약 최소비용의 업체가 여러개라면 cities 배열에서 인덱스 번호가 작은 도시를 답으로 합니다.

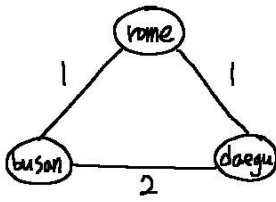
입출력 예

cities	roads	cars	customer	answer
["rome", "busan", "daegu"]	["rome busan 1", "rome daegu 1", "busan daegu 2"]	["rome 50 10", "busan 100, 20", "daegu 40 8"]	"busan daegu 40"	rome

제한사항

- cities의 길이는 1,000을 넘지 않고, 원소는 알파벳 소문자로 이루어져 있으면 길이는 10을 넘지 않습니다. 도시의 이름이 같은 도시는 존재하지 않습니다.
- roads의 길이는 500을 넘지 않으며, 원소는 "a b c"로 이루어져 있고 a도시와 b도시를 양방향으로 연결하는 길이가 c km인 도로를 의미합니다. c의 값을 100을 넘지 않는 자연수입니다. 두 도시를 직접 연결하는 도로는 오직 하나입니다.
- cars의 길이는 1,000,000을 넘지 않으며, 원소는 "a b c"로 이루어져 있고 a는 차량을 가지고 있는 이삿짐센터의 도시이름, b는 이 차량이 운반 할 수있는 최대 무게, c는 1km당 이사비용입니다. $1 \leq b, c \leq 10,000$. 각 이삿짐 센터에는 운반할 수 있는 최대무게가 같은 차량은 존재하지 않습니다. a는 모든 도시의 이름이 한 번 이상 입력됩니다.
- customer는 "a b c"로 이루어져 있고, a는 고객이 사는 출발도시, b는 도착도시, c는 이사하는 총 짐의 무게입니다. c는 10,000이하의 자연수입니다. c는 차 한 대로 이사할 수 있는 무게가 입력됩니다. 차량의 운반 할 수 있는 최대무게가 c이상이면 해당 차량으로 이사할 수 있습니다.

입력예제 1번 설명 :



- 1) rome에 있는 이삿짐 센터는 rome → busan → daegu로 이동하며 최단이동거리는 3이고, 비용은 $3 * 10 = 30$ 입니다.
 - 2) busan에 있는 이삿짐 센터는 busan → daegu로 이동하면 최단이동거리는 2이고, 비용은 $2 * 20 = 40$ 입니다.
 - 3) daegu에 있는 이삿짐 센터는 daegu → busan → daegu로 이동하며 최단이동거리는 4이고, 비용은 $4 * 8 = 32$
- 답은 rome입니다.