

문제 F. 미로 탈출하기

시간 제한 2 초 메모리 제한 1024 MB

$N \times M$ 크기의 미로가 있다. 위에서 i 번째, 왼쪽에서 j 번째 칸을 (i, j) 로 나타낸다. 각 칸은 빈 칸이거나 고정된 벽이며, 미로의 바깥은 모두 벽이다.

미로 안에는 1×1 크기의 움직이는 벽이 K 개 있다. 움직이는 벽 하나는 다섯 정수 x_1, y_1, x_2, y_2, p 로 주어지며, 다음과 같이 움직인다.

- 이 벽은 칸 (x_1, y_1) 과 칸 (x_2, y_2) 를 잇는 직선 구간을 1초에 한 칸씩 왕복한다. $x_1 = x_2$ 또는 $y_1 = y_2$ 가 항상 성립한다.
- 구간의 끝에 도달하면 곧바로 방향을 바꾼다. 방향을 바꾸는 데 걸리는 시간은 없다.
- 시각 0에 이 벽은 (x_1, y_1) 에서 출발한 지 p 초가 지난 위치에 있다. 즉 $d = |x_1 - x_2| + |y_1 - y_2|$ 이고 $s = (p + t) \bmod 2d$ 일 때, 시각 t 에서 이 벽의 위치는 $s \leq d$ 이면 (x_1, y_1) 에서 (x_2, y_2) 쪽으로 s 칸 간 칸이고, $s > d$ 이면 (x_1, y_1) 에서 (x_2, y_2) 쪽으로 $2d - s$ 칸 간 칸이다.

당신은 시각 0에 $(1, 1)$ 에 있고, (N, M) 에 도착해야 한다. 매 초 상하좌우로 인접한 칸으로 한 칸 이동하거나 제자리에 머무를 수 있으며, 두 경우 모두 1초가 걸린다.

시각 t 에 칸 A 에 있다가 시각 $t+1$ 에 칸 B 로 가려고 한다. (B 는 A 이거나 A 와 인접한 칸이다.) 이때 다음 두 조건을 모두 만족해야 한다.

1. 시각 $t+1$ 에 칸 B 가 벽이 아니어야 한다. 즉 B 는 미로 안의 고정된 벽이 아닌 칸이어야 하고, 시각 $t+1$ 에 어떤 움직이는 벽도 B 에 있으면 안 된다.
2. 시각 t 에 칸 B 에 있던 움직이는 벽이 시각 $t+1$ 에 칸 A 에 있으면 안 된다. 즉 움직이는 벽과 자리를 맞바꿀 수 없다.

조건 1의 기준 시각이 출발 시각 t 가 아니라 도착 시각 $t+1$ 이라는 점에 주의한다. 따라서 다음이 성립한다.

- 시각 t 에 어떤 움직이는 벽이 B 에 있었더라도, 시각 $t+1$ 에 그 벽이 B 를 떠났다면 B 로 이동할 수 있다.
- 당신이 떠난 칸 A 가 시각 $t+1$ 에 움직이는 벽으로 채워지는 것은 상관없다.
- 위 두 가지가 동시에 일어나더라도, 관련된 두 움직이는 벽이 서로 다르다면 이동할 수 있다. 같은 벽 하나가 B 에서 A 로 움직인 경우에만 조건 2에 걸린다.

(N, M) 에 도착할 수 있는 가장 이른 시각을 구하여라.

입력

첫 번째 줄에 N, M, K 가 공백으로 구분되어 주어진다.

두 번째 줄부터 N 개의 줄에 걸쳐 미로가 주어진다. 각 줄은 길이가 M 인 문자열이며, 0은 빈 칸, 1은 고정된 벽을 뜻한다.

다음 K 개의 줄에 걸쳐 움직이는 벽의 정보가 주어진다. 각 줄에는 다섯 정수 x_1, y_1, x_2, y_2, p 가 공백으로 구분되어 주어진다.

출력

(N, M) 에 도착할 수 있는 가장 이른 시각을 출력한다. 도착할 수 없다면 -1 을 출력한다.

제한

- $2 \leq N, M \leq 50$
- $1 \leq K \leq 100$
- $1 \leq x_1, x_2 \leq N, 1 \leq y_1, y_2 \leq M$
- $x_1 = x_2$ 또는 $y_1 = y_2$
- $1 \leq d = |x_1 - x_2| + |y_1 - y_2| \leq 10$
- $0 \leq p < 2d$
- $(1, 1)$ 과 (N, M) 은 빈 칸
- 움직이는 벽의 이동 경로에 있는 모든 칸은 빈 칸
- 서로 다른 두 움직이는 벽의 이동 경로는 한 칸도 겹치지 않음
- 어떤 움직이는 벽의 이동 경로도 $(1, 1)$ 과 (N, M) 을 포함하지 않음

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
3 3 1 000 000 000 3 2 1 2 1	5
5 3 2 000 010 010 010 000 2 1 4 1 0 2 3 4 3 0	-1
4 3 3 000 000 000 000 4 1 3 1 0 2 2 2 1 0 2 3 1 3 0	6

노트

예제 1. 움직이는 벽은 2열을 위아래로 오간다. 시각별 위치는 $(2, 2), (1, 2), (2, 2), (3, 2), (2, 2), (1, 2), \dots$ 이다. 2열이 1열과 3열을 가로막고 있으므로 반드시 벽이 비켜난 순간에 2열을 지나가야 한다.

시각	나	움직이는 벽
0	(1, 1)	(2, 2)
1	(1, 1)	(1, 2)
2	(2, 1)	(2, 2)
3	(3, 1)	(3, 2)
4	(3, 2)	(2, 2)
5	(3, 3)	(1, 2)

시각 1에서는 제자리에 머무른다. 시각 3에서 4로 갈 때 (3, 2)에 있던 벽이 (2, 2)로 떠나는 순간 그 칸으로 들어간다. 움직이는 벽을 무시하면 4초면 충분하지만, 실제로는 5초가 걸린다.

예제 2. 1열과 3열이 각각 폭 1짜리 통로이고, 두 통로 안에 움직이는 벽이 하나씩 들어 있다. 폭 1짜리 통로에 있는 움직이는 벽은 앞질러 갈 수도 없고(벽이 곧 되돌아온다) 자리를 맞바꿀 수도 없으므로 절대 통과할 수 없다. 따라서 (5, 3)에 도달할 수 없다.

예제 3. 세 움직이는 벽은 모두 두 칸을 오간다. 시각이 짝수일 때는 (4, 1), (2, 2), (2, 3)에, 홀수일 때는 (3, 1), (2, 1), (1, 3)에 있다.

맞바꿈이 허용된다면 $(1, 1) \rightarrow (1, 2) \rightarrow (1, 3) \rightarrow (2, 3) \rightarrow (3, 3) \rightarrow (4, 3)$ 으로 5초 만에 도착할 수 있다. 그러나 시각 2에서 3으로 갈 때 세 번째 벽이 (2, 3)에서 (1, 3)으로 움직이므로, $(1, 3) \rightarrow (2, 3)$ 이동은 이 벽과 자리를 맞바꾸는 것이 되어 조건 2에 걸린다.

실제 최적 경로는 $(1, 1) \rightarrow (1, 1) \rightarrow (1, 2) \rightarrow (2, 2) \rightarrow (3, 2) \rightarrow (4, 2) \rightarrow (4, 3)$ 으로 6초가 걸린다.