

그래프와 탐색3

🕒 Created	@November 25, 2021 3:24 PM
🏷 Tags	
📁 난이도	

<https://www.acmicpc.net/problem/2206>

벽 부수고 이동하기

문제

$N \times M$ 의 행렬로 표현되는 맵이 있다. 맵에서 0은 이동할 수 있는 곳을 나타내고, 1은 이동할 수 없는 벽이 있는 곳을 나타낸다. 당신은 (1, 1)에서 (N, M)의 위치까지 이동하려 하는데, 이때 최단 경로로 이동하려 한다. 최단경로는 맵에서 가장 적은 개수의 칸을 지나는 경로를 말하는데, 이때 시작하는 칸과 끝나는 칸도 포함해서 센다.

만약에 이동하는 도중에 한 개의 벽을 부수고 이동하는 것이 좀 더 경로가 짧아진다면, 벽을 한 개 까지 부수고 이동하여도 된다.

한 칸에서 이동할 수 있는 칸은 상하좌우로 인접한 칸이다.

맵이 주어졌을 때, 최단 경로를 구해 내는 프로그램을 작성하시오.

예제 입력 1 복사

```
6 4
0100
1110
1000
0000
0111
0000
```

예제 출력 1 복사

```
15
```

예제 입력 2 복사

```
4 4
0111
1111
1111
1110
```

예제 출력 2 복사

```
-1
```

이전에 있었던 연구소 문제 방식으로는 안풀린다.

처음엔 visit[]만 유지했다.
 그리고 Queue에 벽을 부순 상태인지 아닌지를 체크했다.
 이 방법은 틀렸다.



반례를 살펴보자

<https://www.acmicpc.net/problem/2665>

미로 만들기

문제

$n \times n$ 바둑판 모양으로 총 n^2 개의 방이 있다. 일부분은 검은 방이고 나머지는 모두 흰 방이다. 검은 방은 사면이 벽으로 싸여 있어 들어갈 수 없다. 서로 붙어 있는 두 개의 흰 방 사이에는 문이 있어서 지나다닐 수 있다. 윗줄 맨 왼쪽 방은 시작방으로서 항상 흰 방이고, 아랫줄 맨 오른쪽 방은 끝방으로서 역시 흰 방이다.

시작방에서 출발하여 길을 찾아서 끝방으로 가는 것이 목적인데, 아래 그림의 경우에는 시작방에서 끝 방으로 갈 수가 없다. 부득이 검은 방 몇 개를 흰 방으로 바꾸어야 하는데 되도록 적은 수의 방의 색을 바꾸고 싶다.

아래 그림은 $n=8$ 인 경우의 한 예이다.

	1	2	3	4	5	6	7	8
1	시작							
2								
3								
4								
5								
6								
7								
8								끝방

위 그림에서는 두 개의 검은 방(예를 들어 (4,4)의 방과 (7,8)의 방)을 흰 방으로 바꾸면, 시작방에서 끝방으로 갈 수 있지만, 어느 검은 방 하나만을 흰 방으로 바꾸어서는 불가능하다. 검은 방에서 흰 방으로 바꾸어야 할 최소의 수를 구하는 프로그램을 작성하시오.

단, 검은 방을 하나도 흰방으로 바꾸지 않아도 되는 경우는 0이 답이다.

1. 모든 좌표를 INF 로 초기화

2. 시작점부터 검은 색으로 바꾼 숫자를 센다.
3. 내가 검은 색으로 바꾼 숫자가 다음 좌표보다 크면 진행하지 않는다.

<https://www.acmicpc.net/problem/16946>

벽 부수고 이동하기 4

예제 입력 1 복사

```
3 3
101
010
101
```

예제 출력 1 복사

```
303
050
303
```

벽을 부수고 0의 크기를 세야한다.

완전탐색

부술 때 마다 센다.

1,000^1,000 이기 때문에 시간초과!

미리 세두자

1. 미리 세고 더하기
2. 중복 발생!
3. 어떻게 같은 녀석이라고 식별할까? 그룹핑하기

```
4 5
11001
00111
01010
10101
```

```

4 5
11001
00111
01010
10101

```

```

▼ f group = {int[4][]@626}
> 0 = {int[5]@629} [0, 0, 1, 1, 0]
> 1 = {int[5]@630} [2, 2, 0, 0, 0]
> 2 = {int[5]@631} [2, 0, 3, 0, 4]
> 3 = {int[5]@632} [0, 5, 0, 6, 0]

```

```

▼ f groupMap = {HashMap@627} size = 6
> {Integer@641} 1 -> {Integer@642} 2
> {Integer@642} 2 -> {Integer@643} 3
> {Integer@643} 3 -> {Integer@641} 1
> {Integer@644} 4 -> {Integer@641} 1
> {Integer@645} 5 -> {Integer@641} 1
> {Integer@646} 6 -> {Integer@641} 1

```