

# B0J13460 구슬탈출 2

역방향 BFS. 비트마스크 푸는 방법도 있고  
BFS 풀어도 가능. 재귀도 가능 (16197과 비슷)

구멍더 어렵

재귀가 끝나는 경우

재귀로 풀기

① 불가능한 경우

구슬의 이동이 11번 이상임, 파란구슬이 구멍에 빠짐  
파란구슬과 빨간구슬이 동시에 빠짐.

② 가능한 경우

빨간 구슬만 구멍으로 떨어짐 → 빨간구슬이 구멍 (hx, hy)로 떨어지면  
보드를 가로인 횟수 리턴

③ 다음 경우

네 방향으로 가림. 벽을 만날 때까지 이동

다음 경우를 호출할 때 전달해야 함

→ 빨간공 위치 (rx, ry), 파란공 위치 (bx, by),

보드를 가로인 횟수 (cnt)

다음 경우

빨간공이 움직임

벽에 부딪힘 - 파란공 움직임

벽에 부딪힘. 재귀 호출

구멍에 빠짐. 끝

빨간공이 부딪힘. 재귀 호출

벽 충돌. 다음 재귀

구멍에 빠짐. 끝

구멍에 빠짐 - 파란공 move

파란공이 부딪힘 - 파란공 움직임

벽 충돌. 재귀 호출

구멍에 빠짐. 끝

다음 재귀 호출

void go(int rx, int ry, int bx, int by, int cnt)

→ 1대1 구현, 다른 방법 더 생각해보기

보드의 세로 크기는 N, 가로 크기는 M이고, 편의상 1x1크기의 칸으로 나누어져 있다. 가장 바깥 행과 열은 모두 막혀져 있고, 보드에는 구멍이 하나 있다. 빨간 구슬과 파란 구슬의 크기는 보드에서 1x1크기의 칸을 가득 채우는 사이즈이고, 각각 하나씩 들어가 있다. 게임의 목표는 빨간 구슬을 구멍을 통해서 빼내는 것이다. 이때, 파란 구슬이 구멍에 들어가면 안 된다.

→ fall1 == true && fall2 == true

이때, 구슬을 손으로 건드릴 수는 없고, 중력을 이용해서 이리 저리 굴려야 한다. 왼쪽으로 기울이기, 오른쪽으로 기울이기, 위쪽으로 기울이기, 아래쪽으로 기울이기와 같은 네 가지 동작이 가능하다.

→ fall1 == true → 빨간구슬이 구멍으로 떨어짐

각각의 동작에서 공은 동시에 움직인다. 빨간 구슬이 구멍에 빠지면 성공이지만, 파란 구슬이 구멍에 빠지면 실패이다. 빨간 구슬과 파란 구슬이 동시에 구멍에 빠져도 실패이다. 빨간 구슬과 파란 구슬은 동시에 같은 칸에 있을 수 없다. 또, 빨간 구슬과 파란 구슬의 크기는 한 칸을 모두 차지한다. 기울이는 동작을 그만하는 것은 더 이상 구슬이 움직이지 않을 때 까지이다.

→ 한 방향으로 계속 움직임

보드의 상태가 주어졌을 때, 최소 몇 번 만에 빨간 구슬을 구멍을 통해 빼낼 수 있는지 구하는 프로그램을 작성하시오.

## 입력

첫 번째 줄에는 보드의 세로, 가로 크기를 의미하는 두 정수 N, M ( $3 \leq N, M \leq 10$ )이 주어진다. 다음 N개의 줄에 보드의 모양을 나타내는 길이 M의 문자열이 주어진다. 이 문자열은 '.', '#', 'O', 'R', 'B'로 이루어져 있다. '.'은 빈 칸을 의미하고, '#'은 공이 이동할 수 없는 장애물 또는 벽을 의미하며, 'O'는 구멍의 위치를 의미한다. 'R'은 빨간 구슬의 위치, 'B'는 파란 구슬의 위치이다.

입력되는 모든 보드의 가장자리에는 모두 '#'이 있다. 구멍의 개수는 한 개이며, 빨간 구슬과 파란 구슬은 항상 1개가 주어진다.

## 출력

최소 몇 번 만에 빨간 구슬을 구멍을 통해 빼낼 수 있는지 출력한다. 만약, 10번 이하로 움직여서 빨간 구슬을 구멍을 통해 빼낼 수 없으면 -1을 출력한다.

→ 재귀 탈출 조건