

# 마법 동굴

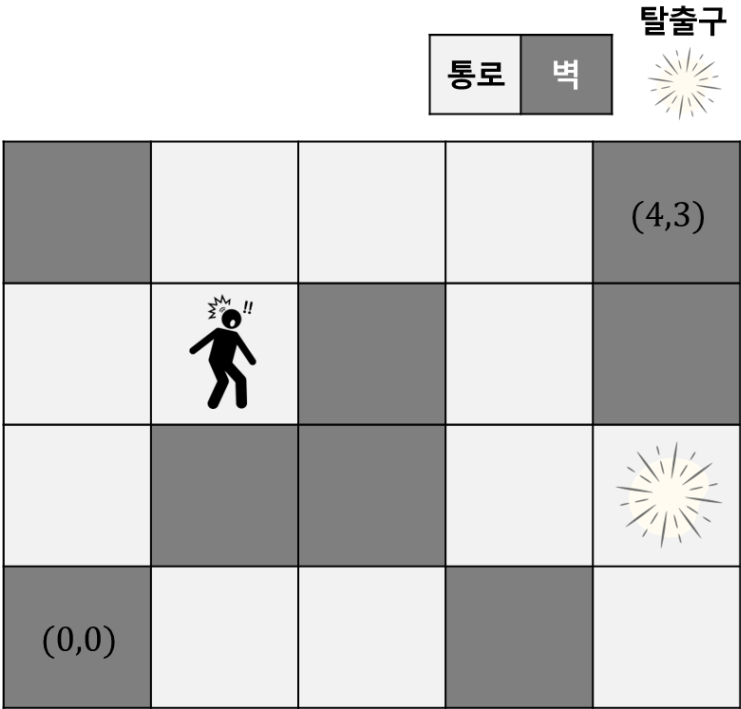
시간 제한	1초
메모리 제한	512 MB
사용 가능 언어	C, C++
C++ 허용 STL	전체

C++의 경우 main 함수 내의 시작 지점에 다음 내용을 추가하여 cin, cout 의 입출력 속도를 개선할 수 있다.

```
ios_base::sync_with_stdio(false);  
cin.tie(nullptr);  
cout.tie(nullptr);
```

단, 위 내용을 추가할 경우 cin, cout 만 사용해야 하며, scanf, printf 등 C 입출력과 혼용해서 사용하면 안 된다. C++ std::endl의 경우 출력 속도가 느리므로, cout<<endl; 대신 **cout<<"\n";**을 사용하는 것을 권장한다.

탐험가 정석이는 오지를 탐험하던 도중 소나기를 피하다가 마법 동굴에 들어가게 되었다. 동굴 안이 신기했던 정석이는 내부를 살펴보며 점점 깊숙하게 들어가다가 벽에 있던 장치를 만지게 되었는데, 굉음과 함께 동굴 내부의 구조가 바뀌었다. 바뀐 동굴의 구조는 위에서 보았을 때 가로 길이가  $N$ 이고 세로 길이가  $M$ 인 2차원 격자 형태이며, 각 격자는 사람이 지나갈 수 있는 열린 칸이거나, 벽으로 막혀 있어 지나갈 수 없는 칸이다. 바뀐 동굴의 구조와 정석이의 현재 위치가 주어졌을 때, 정석이가 마법 동굴에서 탈출할 수 있는지, 만약 그렇다면 탈출을 위해 최소 몇 칸을 거쳐가야 하는지 알아보자. 단, 정석이는 상, 하, 좌, 우 4방향으로만 이동이 가능하며, 대각선 방향으로의 이동이 불가능하다. 동굴의 구조는 편의상 좌표로 주어지며, 가장 왼쪽 아래 격자를  $(0,0)$ , 가장 오른쪽 위 격자를  $(N-1,M-1)$ 으로 표현한다. 동굴의 탈출구는 오직 하나이며, 열린 칸에 위치해 있다.



# 입력

첫 번째 줄에는 테스트 케이스 수  $T$  ( $1 \leq T \leq 50$ )가 주어진다. 이후 각 테스트 케이스의 정보가 다음과 같이 주어진다.

- 첫 번째 줄에 바뀐 동굴의 가로 길이  $N$  ( $1 \leq N \leq 1,000$ ), 세로 길이  $M$  ( $1 \leq M \leq 1,000$ )이 공백으로 구분되어 주어진다.
- 두 번째 줄에 정석이의 좌표  $a,b$  ( $0 \leq a \leq N-1, 0 \leq b \leq M-1$ )와 탈출구의 좌표  $c,d$  ( $0 \leq c \leq N-1, 0 \leq d \leq M-1$ )가 공백으로 구분되어 주어진다.
- 세 번째 줄부터  $M$ 개의 줄에 걸쳐, 동굴의 격자별 정보를 의미하는  $N$ 개의 문자가 공백으로 구분되어 주어진다. 문자는 'X' 또는 'O'로 주어지며, 'X'는 벽으로 막혀 있어 지나갈 수 없는 칸을, 'O'는 열린 칸을 의미한다.

# 출력

각 테스트 케이스마다, 탈출이 가능할 경우 이동해야 하는 격자의 최소 개수(정석이의 시작 위치는 제외하고, 탈출구는 포함함)를 출력하고, 탈출이 불가능할 경우 -1을 한 줄씩 출력한다.

# 예제 입출력

예제 입력	예제 출력
2 5 4 1 2 4 1 X O O O X O O X O X O X X O O X O O X O 3 4 0 3 2 3 O X O O O X O O O O O O	6 -1