

16000번 - 섬

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
5 초 (언어별 추가 시간 없음)	768 MB	485	85	46	23.834%

문제

자가용 비행기로 태평양 한복판을 비행하던 튜브는, 연료 부족으로 외딴섬 한가운데에 불시착하고 말았다! 다행히도, 튜브는 섬 근방의 지역에 대한 정보가 전부 나와있는 $N \times M$ 격자 모양의 지도를 발견하였다.

튜브가 발견한 지도에서, 격자의 각 셀은 전부 바다이거나 육지이며, 최외곽에 있는 셀들은 모두 바다이다. 엄밀하게 정의해서, 격자의 각 셀은 (x, y) ($1 \leq x \leq N, 1 \leq y \leq M$) 형태의 좌표로 표현되며, $x = 1, x = N, y = 1, y = M$ 중 하나의 식이라도 만족하는 셀 (x, y) 는 바다임이 보장된다. 모든 육지 셀은 어떠한 섬의 일부분인데, 두 육지 셀이 인접하다면 (한 면을 맞닿고 있다면) 이들은 하나의 섬으로 간주된다. 다른 말로, (x, y) 와 $(x+1, y)$ 가 모두 육지 셀이면 둘은 같은 섬에 속하고, (x, y) 와 $(x, y+1)$ 이 모두 육지 셀이면 이 둘 역시 같은 섬에 속한다.

튜브는 현재 위치에서 뗏목을 주웠고, 이 뗏목을 타고 저 멀리 망망대해로 먼 여정을 떠날 예정이다. 튜브는 임의의 셀에서 인접한 4방향의 셀로 이동할 수 있으며, 이동 시에는 해당 셀이 육지인지 바다인지의 여부는 중요하지 않다 (뗏목을 주웠으니깐!). 튜브의 목표는 현재의 위치에서 이러한 이동을 통해 저 멀리 바다에서 연료를 찾아 돌아오는 것이다.

하지만 튜브는 두 가지 문제점에 봉착해 있다. 첫 번째 문제는, 튜브가 불시착한 지역에 있는 섬 중 하나의 섬에는 오리를 잡아먹으려 하는 무서운 사냥꾼들이 살고 있다. 튜브가 현재 있는 섬은 안전함을 확인했으나, 만약 탈출 시 사냥꾼이 있는 섬을 지나야만 한다면 튜브는 큰 위험에 처하게 된다! 두 번째 문제는, 튜브가 현재 지도 상의 어느 섬에 있는지를 알지 못한다는 것이다. 튜브가 있는 섬은 최외곽의 바다와 바로 인접해있는 편리한 곳일 수도 있으나, 다른 섬들에 층층이 둘러싸인 위험한 곳일 수도 있다.

튜브는 지도상의 모든 섬에 대해서 해당 섬이 안전한지 위험한지를 알고 싶어 한다. 어떠한 섬 S 가 위험하다는 것은, S 에서 최외곽 셀로 가기 위해서 무조건 지나야 하는 S 와는 다른 섬 T 가 존재한다는 것이고, S 가 안전하다는 것은 S 가 위험하지 않다는 것을 뜻한다. 튜브를 도와서, 지도상에 있는 섬들 중 안전한 섬이 무엇이고 위험한 섬이 무엇인지 알려주자!

입력

첫 번째 줄에는 격자의 크기 N, M 이 공백으로 구분되어 주어진다. ($3 \leq N, M \leq 2,000$)

이후 N 개의 줄에 길이 M 의 문자열이 주어진다. 이 중 i 번째 줄의 j 번째 문자는 (i, j) 셀의 상태를 나타내는데, 만약 이 문자가 '#'이라면 해당 셀은 육지이고, '.'이라면 해당 셀은 바다이다. 문자열 내에서 이 외의 문자는 주어지지 않음이 보장된다.

격자 안에는 하나 이상의 육지 셀이 존재함이 보장된다.

출력

N 개의 줄에 길이 M 의 문자열을 출력하라. 이 중 i 번째 줄의 j 번째 문자는 (i, j) 셀의 상태를 나타내어야 한다. 만약 (i, j) 셀이 바다라면 해당 위치에 '.'을 출력해야 하고, 안전한 섬의 일부라면 해당 위치에 'o'를 출력해야 하고, 위험한 섬의 일부라면 해당 위치에 'x'를 출력해야 한다.

예제 입력 1 복사

```
3 3
...
.#.
...
```

예제 출력 1 복사

```
...
.O.
...
```

1번 예제에서는 섬이 단 하나밖에 없다. 바로 최외곽으로 나갈 수 있다. 참고로, 이 입력은 3 * 3 격자로 가능한 유일한 입력이다. 8개의 셀은 최외곽 셀이기 때문에 무조건 바다여야 하고, 나머지 하나의 셀은 무조건 육지여야 모든 입력이 바다가 아니기 때문이다.

예제 입력 2 복사

```
9 7
.....
.#####.
.#...#.
.#...#.
.#...#.
.###.###.
.#...#.
.#...#.
.#####.
.....
```

예제 출력 2 복사

```
.....
.00000.
.O...O.
.O...O.
.O.X.O.
.O...O.
.O...O.
.O...O.
.00000.
.....
```

2번 예제에서는 큰 섬 하나가 작은 섬 하나를 둘러싸고 있는 모양이다. 큰 섬에서는 바로 최외곽으로 나갈 수 있으나, 작은 섬에서는 무조건 큰 섬을 거쳐야만 최외곽으로 나갈 수 있다. 고로 큰 섬은 안전하고, 작은 섬은 위험하다.

예제 입력 3 복사

```
9 7
.....
..###..
.#...#.
.#...#.
.###.###.
.#...#.
.#...#.
..###..
.....
```

예제 출력 3 복사

```
.....
..000..
.0...0.
.0...0.
.0.0.0.
.0...0.
.0...0.
..000..
.....
```

3번 예제에서는 큰 섬 4개가 작은 섬 하나를 둘러싸고 있는 모양이다. 큰 섬에서는 바로 최외곽으로 나갈 수 있으니, 안전함이 자명하다. 둘러싸인 작은 섬에서는 무조건 지나야 하는 하나의 큰 섬은 없다 - 즉, 만약 이 중 하나에 사냥꾼이 살고 있으면, 그냥 다른 섬으로 돌아가면 된다. 고로 작은 섬도 안전하다.

출처

Contest (/category/45) > 카카오 코드 페스티벌 (/category/428) > 카카오 코드 페스티벌 2018 (/category/detail/1915) D번