

- 문제

- 좌표의 크기가 주어지고, 문명 발생지의 수와 각 좌표값이 주어진다.

* 주의해야 할 조건

① 최초 발생지끼리 인접 → 처음부터 하나로.

② "만약 두 인접하는 지역에 다른 문명이 전파되었거나, 한 지역에 둘 이상의 다른 문명이 전파된다면 두 문명은 결합된다."

- 접근

① vector에 모든 좌표 넣고 1씩 확장 → 연결요소 개수 체크 (시간초과)

② BFS로 날짜마다 $Q.size()$ 만큼 돌면서 ^{먼저 변수에 넣고 돌리지 않으면 개량함.} 확장시킨 후 DFS로 같은 숫자로 바꿔줌 (정복 ++). (시간초과).

↓
정복의 표시를 더 빠르게 해줄 필요가 있음.

③ union-find를 통해 노드의 번호가 달라도 parent가 같으면 같은 노드로 친다.

- 구현

· 조건 ① : 좌표에 발생지 넣을 때마다 네 방향 체크.

· 조건 ② : "한 지역에 두 문명이 전파"는 "+1씩 확장하는 bfs 탐색 중"을 의미 하지만

"인접한 곳에 다른 문명"은 "+1 확장 후끼리 동서남북에 인접해 있음"을 의미하기 때문에

⇒ " " 는 bfs 탐색 중 [ny][nx]가 '.'가 아닌 다른 문명일 때 체크하면 되고

" " 는 [ny][nx]가 빈 땅인 '.' 인 +1 확장하고 나서 한번 더 4방향을 검사해서 인접한지를 확인해야 한다.

- 정복 : Find 함수를 통해 parent가 다르다면 Union하면서 conquer ++ 해준다. (countries - 1 이되면 하나의 문명이 된 것).

* 주의해야 할 점.

1. day를 나타내기 위해 q.size() 만큼씩만 돌릴 때.

int len = q.size(); 변수에 이렇게 넣고서
돌리지 않으면 for (int i = 0; i < q.size(); ++i)

↑
큰일난다!!

2. 문제에 조건이 많으므로 (정복수 + 1 == 문명수) 를 부분부분 잘 체크한다. (입력 시기도 인접 좌표에 반상지 생기면 바로 정복 + 1.)

3. 처음부터 문명이 1개면 0 출력 → 종료.

(입력을 마친 후 (정복 + 1 == 문명수) 체크하면 됨.

처음부터 1개면 어이 == 1, 처음부터 다 붙어있으면 3 + 1 = 4.)