

[22.06.10] 섬지키기

TS섬지키기

김 태 현



문 제

N*N 크기 섬이 바다에 둘러싸인 상태로 존재

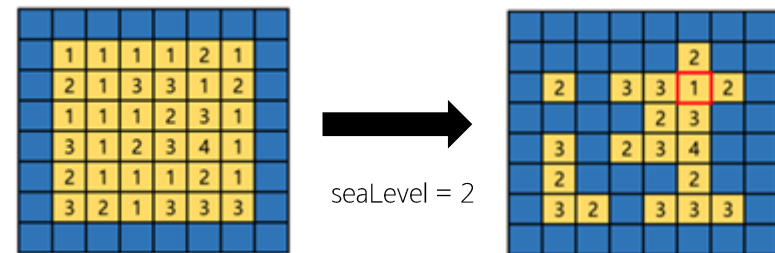
각 위치마다 고도를 갖고 있으며 해수면이 seaLevel 이면, seaLevel-1 이하인 지역으로 침투 가능

(상하좌우 4방향으로만 침투)

별개의 높이를 갖는 1*M 크기의 구조물 존재

구조물은 90, 180, 270도 회전하여 설치 가능하며, 설치된 M개 지역의 고도는 그 높이만큼 증가

단, 설치한 이후 M개 지역의 높이가 모두 같은 경우에만 설치 가능



```
void init(int N, int land[20][20])
```

N*N 크기의 각 지역 고도 land[0~N-1][0~N-1]

```
int getCount(int M, int structure[5])
```

1*M 크기 구조물을 설치 할 수 있는 경우의 수를 반환

설치 지역이 모두 동일하면, 같은 경우로 취급

```
int getMaxArea(int M, int structure[5], int seaLevel)
```

해수면이 seaLevel일 때, 남은 지역의 개수가 최대가 되도록 구조물을 설치하였을 때의 그 개수를 반환

※ 제약사항

5 ≤ N ≤ 20

1 ≤ M ≤ 5

1 ≤ land[i][j], structure[i] ≤ 5

getCount() ≤ 150,000

getMaxArea() ≤ 50

getMaxArea() 에서 구조물을 설치할 수 있는 경우의 수 총 합 ≤ 5,000

getCount()

Naïve

매 호출시마다 모든 상태가 설치 가능한지 확인

$O(\text{호출횟수} * \text{모든 위치} * \text{방향} * \text{확인비용}) = O(150,000 * 20^2 * 4 * 5)$

Hash

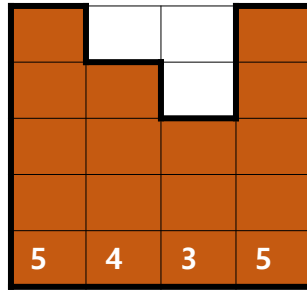
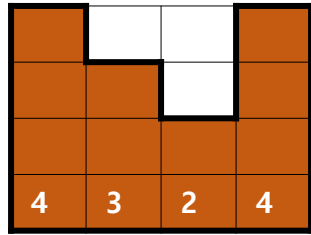
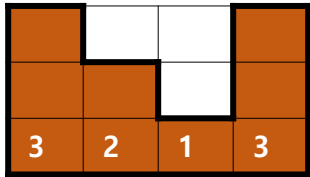
특정 구조물로 설치 가능한 모든 상태들을 미리 분류

Key (땅 상태)	(X,Y,방향)
11	(0,0,우) (2,3,하) (3,1,우) (3,2,우) (2,0,하)
12	(0,2,좌) (1,2,우) (1,0,우) ..
122	(0,2,좌) (3,0,우) ..
123	(2,2,좌) (2,1,상) ..

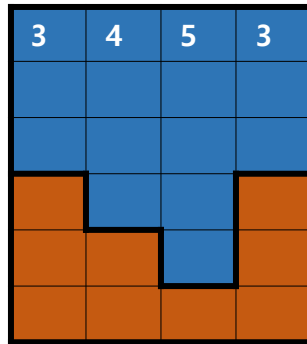
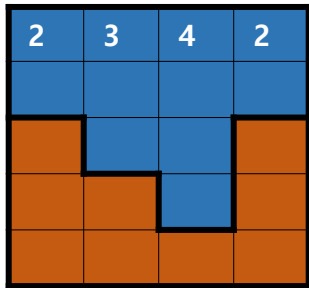
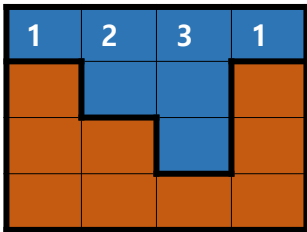
	0	1	2	3
0	1	1	2	1
1	3	2	5	4
2	2	3	4	1
3	2	1	1	1

getCount()

Hash Key 값 설정(1)



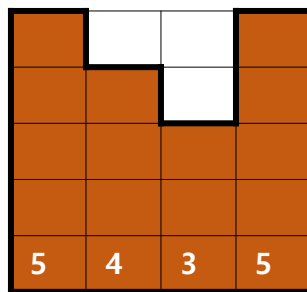
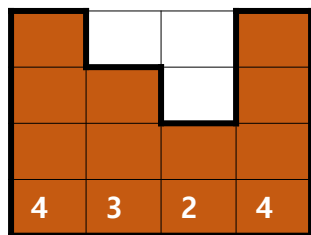
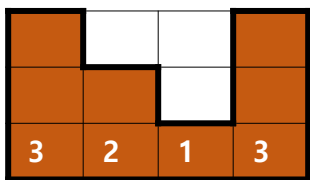
왼쪽 세가지 땅은 같은 구조물을 세울 수 있다.
고도의 절대적인 수치가 아니라 상대적인 수치가 중요
가장 작은 값을 1로 맞추고 10진법 표현하면 통일 가능 : 3213



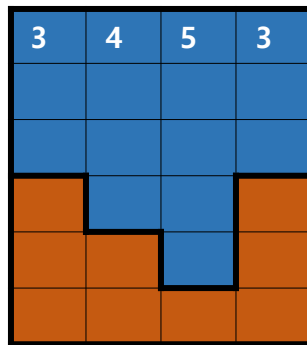
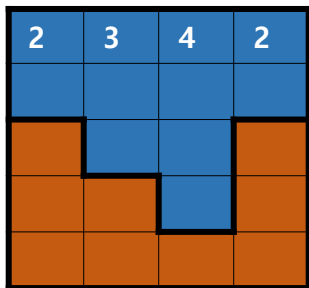
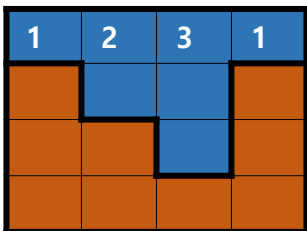
왼쪽 세가지 구조물은 같은 땅에 놓일 수 있다.
(구조물의 가장 큰 높이 + 1) - 높이를 계산하여
10진법 표현하면 놓을 수 있는 땅의 key값이 나옴 : 3213

getCount()

Hash Key 값 설정(2)



차이를 이용하여 key값 설정
(오른쪽 - 왼쪽 + 5) 를 10진법으로
 $(-1, -1, 2) \rightarrow (4, 4, 7) \rightarrow 447$



반대로 (왼쪽 - 오른쪽 + 5) 를 10진법으로 한다.
 $(-1, -1, 3) \rightarrow (4, 4, 7) \rightarrow 447$

getCount()

Hash Key(1) Example

4가지 방향을 중복되지 않게 전부 담는다.

Key (땅 상태)	(X,Y,방향)
11	(0,0,하) , (1,0,우)
3213	(2,3,좌) , (3,3,상) , (0,1,하)
..	

	0	1	2	3
0	3	4	1	5
1	3	3	5	3
2	4	2	3	4
3	2	4	1	5

structure[] = {2, 3, 4, 2} 일 때,
가장 큰 값 + 1 인 5에서 각 값을 빼서 {3, 2, 1, 3} -> key = 3213 의 노드 개수 3 반환

getMaxArea()

전형적인 flood-fill (DFS or BFS)

hash를 통해 설치 가능한 위치는 바로 확인 가능

설치할 수 있는 경우의 수 총합 $\leq 5,000$

$O(\text{경우의 수} * N^2 * 4) = O(5,000 * 20^2 * 4)$

1) 바다와 맞닿아 있는 모든 부분 중 잠기는 부분을 queue에 담고 BFS

	0	1	2	3
0	1	1	2	1
1	3	2	5	4
2	2	3	4	1
3	2	1	1	1

2) (1,1)부터 섬을 표현하고 (0,0) 을 queue에 담고 BFS

	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	1	2	1	0
2	0	3	2	5	4	0
3	0	2	3	4	1	0
4	0	2	1	1	1	0
5	0	0	0	0	0	0

감사합니다

