

- ❖ 학습해야 할 내용
- ✓ Condition- if, switch
- ✓ Loop for, while, do~while
- ✓ Array
- **1. 소금쟁이 중첩** (패키지명 : com.ssafy.algo, 클래스명: Solution21)

N*N의 정사각형 연못이 있다고 가정할 때, 소금쟁이가 연못의 물위를 뛰어 다닌다. 소금쟁이가 처음에 뛸 때는 3칸, 두 번째 뛸 때는 2칸, 세 번째 뛸 때는 1 칸 뛰게 된다. 여러 마리의 소금쟁이가 시작위치와 방향을 달리하여 뛰었을 때연못의 밖으로 나가거나, 앞에서 뛰었던 위치를 또 뛸 수 있다. 앞에서 뛰었던 영역을 또 뛰게 되었을 때 몇 번째 소금쟁이 인지를 출력하세요.

[제한조건]

N은 자연수 5~20 의 값이다.

소금쟁이의 처음 위치와 방향이 주어진다.

처음 소금쟁이는 1번 다음 소금쟁이는 2번 순으로 번호를 부여한다.

여러 마리의 소금쟁이가 순서대로 3번씩 뛴다. 뛸 때 마다 3칸, 2칸 1칸씩 뛴다.

뛰는 도중 연못 밖으로 나가면 그 소금쟁이는 뛰기를 멈춘다.

같은 영역을 두 번 뛰게 되었을 때 소금쟁이 번호를 출력한다.(같은 장소를 두 번 뛴 처음 소금쟁이 번호 출력)

어떤 소금쟁이의 시작 위치가 이미 뛰었던 자리라면 그 소금쟁이의 번호를 출력한다.

같은 자리를 뛴 소금쟁이가 하나도 없다면 0을 출력한다.



[입력]

첫 줄에는 테스트 케이스 T가 주어진다.

두 번째 줄에는 배열의 크기인 N과 소금쟁이 수가 주어진다..

그 다음 줄부터 시작위치(행,열), 방향(하:1, 우:2)가 소금쟁이 수만큼 주어진다.

다음 테스트 케이스가 반복적으로 주어진다.

[출력]

각 줄은 #Ti(테스트케이스 번호) 공백을 하나 둔 다음 같은 곳을 처음 뛴 소금쟁이 번호를 출력한다.

[입력 예]

| 3 // 테스트 케이스 수 | 8 3 2 |
|-------------------------------|-------|
| 9 5 // 연못의 크기 (N*N), 소금쟁이 수 | 7 5 |
| 6 2 2 // 처음위치 (6,2), 오른쪽 방향으로 | 6 2 2 |
| 1 5 1 | 1 6 1 |
| 0 0 2 | 0 0 2 |
| 1 0 1 | 1 0 1 |
| 2 4 2 | 2 4 2 |
| 11 6 // 연못의 크기 (N*N), 소금쟁이 수 | |

| 0 0 2 | [출력 예] |
|-------|--------|
| 1 0 1 | #1 2 |
| 2 4 2 | #2 5 |
| 6 2 2 | #3 0 |
| 151 | |

❖ 제출방법

- 프로젝트명 : "hwjava03_반_성명" 으로 작성
- 완성 후 zip으로 압축하여 제출



2. 미로 도착지점 (패키지명: com.ssafy.algo, 클래스명: Solution42)

N*N의 칸이 있는 미로(미로의 좌표는 (1,1) 부터 시작한다)가 있다. 각 칸에는 동서남북으로 문이 있다. 미로에서 빠져 나오도록 돕기 위해 시작위치와 어떤 방향의 문으로 몇 칸을 이동하라는 숫자를 제공한다. 이동을 하다 보면 밖으로 나올수 있는 점퍼가 있다. 점퍼에 도착하면 밖으로 이동할 수 있다. 점퍼를 이용하여밖으로 나왔거나 이동 시 N*N의 칸을 벗어나면 좌표 값은 (0,0)을 갖게 된다. 주어진 지시(방향과 이동 칸 수)를 모두 수행 했을 때 도착지점을 출력하는 프로그램을 작성하세요.

이동방향 값

| | | 1 | | |
|---|----------|---|---|---|
| 4 | \wedge | = | > | 2 |
| | | 7 | | |

[제한조건]

- ▶ N은 자연수 3~20 의 값이다.
- 출발점 좌표에서 이동 지시의 방향으로 제시된 이동 칸 수만큼 반복적으로 이동한다.
- 이동 시 점퍼에 의해 밖으로 나오거나, N*N 칸을 벗어 났을 경우 좌표는 (0,0)을 갖고 이후 이동 명령은 무시한다.

[입력]

- ▶ 첫 줄에는 테스트 케이스 T(1~100)가 주어진다.
- ➤ 두 번째 줄에는 배열의 크기인 N과 출발점의 좌표(행,열), 점퍼의 개수(1~N)가 주어진다.
- ▶ 세 번째 줄에는 점퍼의 개수만큼 점퍼의 좌표(행,열)가 주어진다..
- ▶ 네 번째 줄에는 방향 지시의 개수(0~50)가 주어진다.
- ▶ 다섯 번째 줄에는 방향 지시의 개수만큼 (방향, 이동칸수)*방향지시개수 가 차례로 주어진다.
- ▶ 다음 테스트 케이스가 반복적으로 주어진다.



[출력]

각 줄은 #Ti(테스트케이스 번호) 공백을 하나 둔 다음 목적지의 좌표를 (행, 열) 순서로 출력한다.

[입력 예]

3 // 테스트 케이스 수

8 5 3 4 // 배열크기, 출발점좌표, 점퍼의 개수

18551267 // 점퍼의 좌표

5 // 이동지시 개수

1321314112 // 이동지시(방향, 이동칸수) * 5

7 4 4 4

17551263

4

43312232

5 1 1 1

1 2

5

3 2 2 1 3 3 2 4 3 1

[출력 예]

#1 1 3

#2 0 0

#3 0 0