[문제11] 로봇 이동거리

Abc 로봇회사는 직진만 할 수 있는 로봇을 만들었다. A 로봇은 우(오른쪽)으로만 갈수 있고, B 로봇은 좌우로 갈수 있으며, C 로봇은 상하좌우로 이동 가능하다. 각 로봇들이 갈 수 있는 거리를 구하세요. N\*N의 네모 칸들로 이루어진 공간이 있고, 각 공간은 비어있거나 벽으로 되어 있다. 벽이 있는 공간을 만나면 로봇들은 더 이상 이동하지 못한다. 다른 로봇이 있던 초기 위치를 넘어가지 못한다고 가정한다.



A(0,1)은 우2칸 이동, A(1,2)은 우1칸 이동 /B(4,1)은 좌1칸, 우2칸 이동

C(1,4)은 좌1칸, 우1칸, 하1칸 이동/로봇들의 이동 가능 칸수 : 9칸

[제한조건]

* N은 자연수 3~20 의 값이다.
* 다른 로봇의 초기 위치는 넘어갈 수 없다.

[입력]

* 첫 줄에는 테스트 케이스 T가 주어진다.
* 다음 줄에는 배열의 크기인 N이 주어진다.
* 각 배열의 값 (S는 공백, W는 벽, A,B,C로봇) 이 배열의 개수 만큼 입력된다.

[출력]

각줄은 #Ti(테스트케이스 번호) 공백을 하나 둔 다음 모든 로봇의 이동거리를 출력한다.

[입력 예]

3

6

S A S S W S

S W A S C S

S W S W S S

S W S S W S

S B S S W S

S S S S S S

3

S S A

B W W

S W C

10

C S S S S S S S S C

S W S S S S S S W S

S S W S S S S W S S

A S S W S S W S S A

S S S S S B S S S S

S S S S B S S S S S

A S S W S S W S S A

S S W S S S S W S S

S W S S S S S S W S

C S S S S S S S S C

[출력 예]

#1 9

#2 0

#3 62

[문제13] 빌딩

신도시에 빌딩들을 세우려고 한다. 신도시를 세울 부지는 변의 길이가 N인 정사각형의 형태이며 이 부지는 가로, 세로 길이가 1인 구획으로 나누어져 있다.

구획당 하나의 빌딩을 세울 수 있고, 빌딩을 세울 수 있는 구획은 B로 표시, 공원 조성단지는 G로 표시되어 있다. 빌딩을 세울 때 인접한 구획에 공원 조성 단지 G가 있다면 2층 높이로 세울 수 있고, 인접한 구획에 공원 조성 단지 G가 없다면 현 위치의 가로 위치에 있는 빌딩구획 B와 세로 위치의 빌딩 구획 B의 수를 더한 크기만큼 빌딩을 세울 수 있다. 가장 높이 세울 수 있는 빌딩은 몇 층인가?



[제한조건]

* N은 자연수 3~20 의 값이다.
* 빌딩의 높이를 결정 짓은 인접구획은 상하좌우뿐만 아니라 대각선 위치도 포함한다.
* 현 위치가 경계일 경우 인접구획 체크는 N\*N 배열 내에 있는 곳만을 대상으로 한다. (N\*N 배열 경계밖에는 G구역이 없다고 본다.)
* 빌딩의 높이는 인접구획에 공원 조성 단지 G가 없을 경우 그 위치의 가로와 세로 위치의 빌딩(B)구획의 수의 합으로 결정한다. 현 위치의 B 구획도 포함한다.

[입력]

* 첫 줄에는 테스트 케이스 T가 주어진다.
* 다음 줄에는 배열의 크기인 N이 주어진다.
* 각 배열의 값들이 빌딩 구획이면 B가, 공원조성단지이면 G가 입력된다.

출력]

각 줄은 #Ti(테스트케이스 번호) 공백을 하나 둔 다음 가장 높은 빌딩 높이를 출력한다.

[입력 예]

3

6

G B G G B B

G B G G B G

B B B B G B

B G B B B B

G B B B B G

G B B B B G

5

G B G G B

G B G G B

B B B B G

B G B B B

G B B B B

3

G G B

G B B

B B B

[출력 예]

#1 7

#2 7

#3 5

[문제21] 소금쟁이 중첩

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 1칸 뛰게 된다. 여러 마리의 소금쟁이가 시작위치와 방향을 달리하여 뛰었을 때 연못의 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 출력

[입력 예]

3 // 테스트 케이스 수

9 5 // 연못의 크기 (N\*N), 소금쟁이 수

6 2 2 // 처음위치 (6,2), 오른쪽 방향으로

1 5 1

0 0 2

1 0 1

2 4 2

11 6 // 연못의 크기 (N\*N), 소금쟁이 수

0 0 2

1 0 1

2 4 2

6 2 2

1 5 1

8 3 2

7 5

6 2 2

1 6 1

0 0 2

1 0 1

2 4 2

[출력 예]

#1 2

#2 5

#3 0

[문제22] 소금쟁이 합계

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 1칸 뛰게 된다. 여러 마리의 소금쟁이가 시작위치와 방향을 달리하여 뛰었을 때 연못의 밖으로 나가거나, 앞에서 뛰었던 소금쟁이가 세 번째 뛰고 머무른 위치와 충돌하면 죽는다. 연못 내에 살아있는 소금쟁이 수를 출력하세요.

[제한조건]

* N은 자연수 5~20 의 값이다.
* 소금쟁이의 처음 위치와 방향이 주어진다.
* 여러 마리의 소금쟁이가 순서대로 3번씩 뛴다. 뛸 때 마다 3칸, 2칸 1칸씩 뛴다.
* 3번 뛴 소금쟁이는 그 자리에 머문다.
* 뛰는 도중 연못 밖으로 나가거나 다른 소금쟁이가 머물러 있는 곳으로 뛰면 죽는다.
* 시작위치에 이미 다른 소금쟁이가 있다면 죽는다.

[입력]

* 첫 줄에는 테스트 케이스 T가 주어진다.
* 두 번째 줄에는 배열의 크기인 N과 소금쟁이 수가 주어진다.
* 그 다음 줄부터 시작위치(행,열), 방향(상:1, 하:2, 좌:3, 우:4)가 소금쟁이 수만큼 주어진다.
* 다음 테스트 케이스가 반복적으로 주어진다.

[출력]

각 줄은 #Ti(테스트케이스 번호) 공백을 하나 둔 다음 연못 내에 살아남은 소금쟁이 수를 출력한다.

[입력 예]

3 // 테스트 케이스 수

9 5 // 연못의 크기 (N\*N), // 소금쟁이 수

6 2 4

1 5 2

0 0 4

6 6 1

2 4 3

11 6

0 0 4

6 0 1

2 4 3

4 2 4

1 5 2

10 8 1

7 5

5 2 1

1 6 2

0 6 3

1 0 2

2 4 4

[출력 예]

#1 3

#2 4

#3 1

[문제32] 점프 사방

Y \* X 직사각형의 놀이 공간이 있다. 이 공간 안에는 특정한 자연수가 적혀 있다. 그 자연수는 두 자리 숫자이고, 앞자리는 방향(동남서북 순으로 1~4까지)을 뒷자리는 점프칸수를 의미한다.

게임 참가자는 참가비 1000원을 지불한다.

참가자는 시작 좌표와 점프 횟수를 부여 받는다.

시작 좌표의 값을 분석하여 (앞자리: 방향, 뒷자리 점프칸수)를 점프 횟수만큼 이동한다.

점프 후 마지막 좌표에 적어진 숫자에 100을 곱한 금액을 상금으로 받게 된다.

테스트 케이스 내에 여러 명의 참가자가 있고, 그 참가자들의 상금의 합을 구하여 출력하라.



[제한조건]

* X와 Y은 자연수 3~20 의 값이다.
* 참가자 N은 1~5사이값이다.
* 숫자판에 숫자가 없고 색칠된 부분은 함정이다.(0으로 표시)
* 점프 시 함정에 빠지거나 경계를 벗어나면 상금을 받기 못한다. 참가비 1000을 내고 상금을 받지 못하면 -1000이 더해진다.
* 참가자에게 주어지는 점프 횟수는 1~100 사이 값이다.

[입력]

* 첫 줄에는 테스트 케이스 T가 주어진다.
* 두 번째 줄에는 배열의 크기인 Y, X, 참가자 수(N)이 주어진다.
* 세 번째 줄부터는 X\*Y의 숫자판 정보가 주어진다.
* 그 다음부터는 참가자의 시작위치 행, 열, 횟수가 참가자 수만큼 주어진다.
* 그 다음엔 함정 수, 함정의 좌표 \* 수로 주어진다.
* 다음 테스트 케이스가 반복적으로 주어진다.

[출력]

각 줄은 #Ti(테스트케이스 번호) 공백을 하나 둔 다음 테스트 케이스 내 참가자들의 상금 합계를 출력한다.

[입력 예]

3

6 6 2

34 42 22 32 21 24

22 32 21 33 33 33

13 11 12 21 43 13

34 23 22 24 32 11

22 24 21 41 41 42

31 42 34 32 15 32

2 3 2

2 6 6

3 3 4 4 2 6 5

2 2 3

11 21

41 31

1 1 2

2 1 2

1 2 1

1 2 2

5 6 2

22 32 21 33 33 33

13 11 12 21 43 13

34 23 22 24 32 11

22 24 21 41 41 42

31 42 34 32 15 32

5 4 2

1 3 1

2 3 4 4 2

[출력 예]

#1 2300

#2 -900

#3 1500

[문제42] 미로 도착지점

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 나올 수 있는 점퍼가 있다. 점퍼에 도착하면 밖으로 이동할 수 있다. 점퍼를 이용하여 밖으로 나왔거나 이동 시 N\*N의 칸을 벗어나면 좌표 값은 (0,0)을 갖게 된다. 주어진 지시(방향과 이동칸수)를 모두 수행 했을 때 도착지점을 출력하는 프로그램을 작성하세요.

이동방향 값

[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

1 8 5 5 1 2 6 7 // 점퍼의 좌표

5 // 이동지시 개수

1 3 2 1 3 1 4 1 1 2 // 이동지시(방향, 이동칸수) \* 5

7 4 4 4

1 7 5 5 1 2 6 3

4

4 3 3 1 2 2 3 2

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