

SWEA 달팽이 숫자

📅 날짜	@2021년 8월 3일 → 2021년 8월 3일
🔗 링크	
📄 열	
🏷 태그	D2 SWEA 탐색

$\begin{matrix} \text{col} & 0 \\ \text{row} & +1 \end{matrix} \rightarrow$
 $\begin{matrix} \text{col} & +1 \\ \text{row} & 0 \end{matrix} \downarrow$
 $\begin{matrix} \text{col} & 0 \\ \text{row} & -1 \end{matrix} \leftarrow$
 $\begin{matrix} \text{col} & -1 \\ \text{row} & 0 \end{matrix} \uparrow$

$dc = \{0 \quad 1 \quad 0 \quad -1\}$

$dr = \{1 \quad 0 \quad -1 \quad 0\}$

입력 $N \times N$ Scanner next line

$\left. \begin{matrix} T \\ N_1 \\ N_2 \\ \vdots \end{matrix} \right\}$
 숫자 하나씩
입력

```

cnt = 1  dir = 0;
while (N^2 > cnt) {
    col += dc[dir];
    row += dr[dir];
    if (tmpC < 0 || (tmpC >= N) || (tmpR < 0)
        || (tmpR > N)) {
        col -= dc[dir % 4];
        row -= dr[dir % 4];
        dir++;
        row += dc[dir % 4];
    }
}
    
```

End 11 [실행]

map [col:row] = cnt++;
}

map 종료

```
package d2;
```

```

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

//class Solution {
public class 달팽이숫자2 {
    public static void main(String args[]) throws IOException {
        int[] dc = { 0, 1, 0, -1 };
        int[] dr = { 1, 0, -1, 0 };
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        int N = Integer.valueOf(br.readLine());
        for (int i = 1; i <= N; i++) {
            int mapSize = Integer.valueOf(br.readLine());
            int[][] map = new int[mapSize][mapSize]; // map make
            for (int c = 0; c < mapSize; c++) { // map init 런타임 에러 혹시 몰라서
                for (int r = 0; r < mapSize; r++) {
                    map[c][r] = 0;
                }
            } // end map init
            int col = 0;
            int row = -1;
            int dir = 0;
            int cnt = 1;
            while ((mapSize * mapSize) >= cnt) { // simul
                col += dc[dir % 4];
                row += dr[dir % 4];
                if (!((col >= 0) && (col < mapSize) && (row >= 0) && (row < mapSize) &
& (map[col][row] == 0))) {
                    col -= dc[dir % 4];
                    row -= dr[dir % 4];
                    dir++;
                    col += dc[dir % 4];
                    row += dr[dir % 4];
                }
                map[col][row] = cnt++;
            } // end simul
            System.out.println("#" + i); // output
            for (int c = 0; c < mapSize; c++) {
                for (int r = 0; r < mapSize; r++) {
                    System.out.print(map[c][r] + " ");
                }
                System.out.println();
            } // end output
        } // end for
        br.close();
    }
}

```