

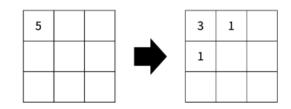
미세먼지 안녕!

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	512 MB	27066	14651	9842	53.428%

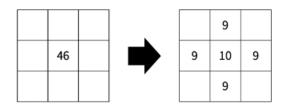
문제

공기청정기는 항상 1번 열에 설치되어 있고, 크기는 두 행을 차지한다. 공기청정기가 설치되어 있지 않은 칸에는 미세먼지가 있고, (r, c)에 있는 미세먼지의 양은 A_{r,c}이다. 1초 동안 아래 적힌 일이 순서대로 일어난다.

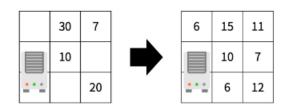
- 1. 미세먼지가 확산된다. 확산은 미세먼지가 있는 모든 칸에서 동시에 일어난다.
 - (r, c)에 있는 미세먼지는 인접한 네 방향으로 확산된다.
 - 인접한 방향에 공기청정기가 있거나, 칸이 없으면 그 방향으로는 확산이 일어나지 않는다.
 - 확산되는 양은 A_{r.c}/5이고 소수점은 버린다.
 - (r, c)에 남은 미세먼지의 양은 A_{r,c} (A_{r,c}/5)×(확산된 방향의 개수) 이다.
- 2. 공기청정기가 작동한다.
 - 공기청정기에서는 바람이 나온다.
 - 위쪽 공기청정기의 바람은 반시계방향으로 순환하고, 아래쪽 공기청정기의 바람은 시계방향으로 순환한다.
 - 바람이 불면 미세먼지가 바람의 방향대로 모두 한 칸씩 이동한다.
 - 공기청정기에서 부는 바람은 미세먼지가 없는 바람이고, 공기청정기로 들어간 미세먼지는 모두 정화된다.



왼쪽과 오른쪽에 칸이 없기 때문에, 두 방향으로만 확산이 일어났다.

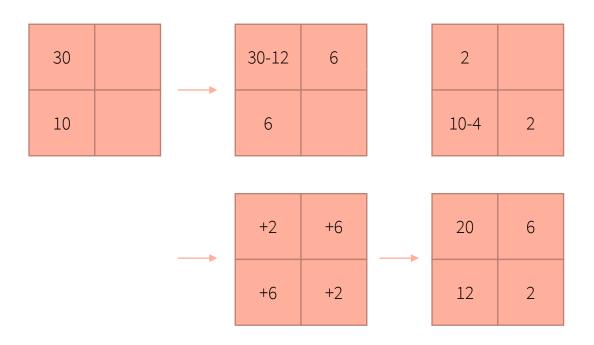


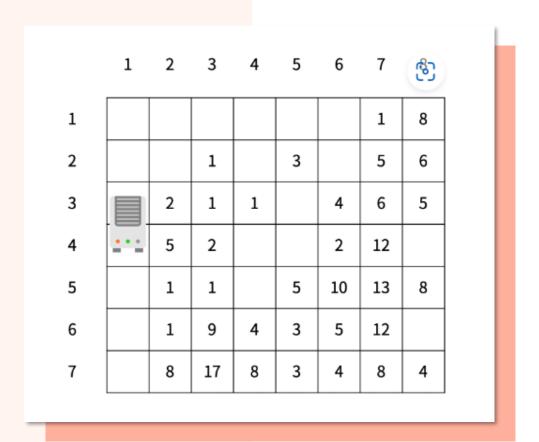
인접한 네 방향으로 모두 확산이 일어난다.

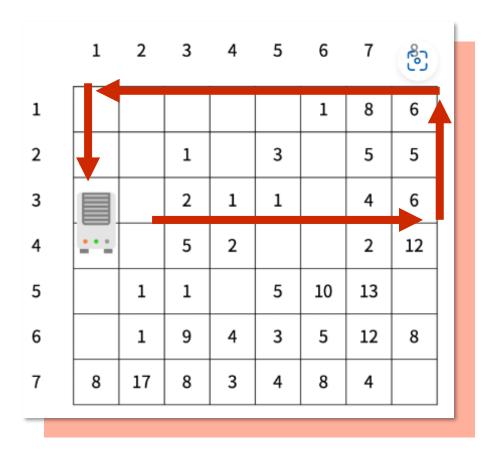


공기청정기가 있는 칸으로는 확산이 일어나지 않는다.

- 1. 미세먼지가 확산된다. 확산은 미세먼지가 있는 모든 칸에서 동시에 일어난다.
 - (r, c)에 있는 미세먼지는 인접한 네 방향으로 확산된다.
 - 인접한 방향에 공기청정기가 있거나, 칸이 없으면 그 방향으로는 확산이 일어나지 않는다.
 - 확산되는 양은 A_{r,c}/5이고 소수점은 버린다.
 - \circ $(r, c)에 남은 미세먼지의 양은 <math>A_{r,c}$ $(A_{r,c}/5) \times (확산된 방향의 개수) 이다.$







바람이 순환하는 방향으로 미세먼지가 한 칸 씩 밀림

구현

확산된 먼지 정보를 기록 - spread() 각 필드에 확산된 미세먼지 값을 newField[51][51]에 기록해둠

미세먼지 값 업데이트 – afterSpread()
newField[51][51]를 바탕으로 원래 필드인 field[51][51]에 업데이트 공기청정기 작동 - clean()
For 문을 통해 공기청정기의 바람에 따라 미세먼지가 한 칸 씩 밀리도록 함

전체 미세먼지 합 - totalFindDust()
For 문을 통해 필드에 남은 미세먼지의
총 합을 구함

```
int main()
   ios base::sync with stdio(false);
    cin.tie(NULL);
    cout.tie(NULL);
   int T;
    cin >> R >> C >> T;
    for(int i=1; i<=R; i++) {
        for(int j=1; j<=C; j++) {
            cin >> field[i][j];
        if (field[i][1] == -1 && airPurifierRoWPos[0] == 0) {
                airPurifierRoWPos[0] = i;
                airPurifierRoWPos[1] = i+1;
   while(T--) {
        spread();
        afterSpread();
        clean();
    cout << totalFindDust() << endl;</pre>
   return 0;
```

For 문을 돌며 필드 채우기

- -1인 필드를 만났다?!
- → 공기청정기 위치
- → 공기청정기는 1열에 연속된 2칸을 차지하므로...

T번 반복

미세먼지 확산(정보 기록)

필드 값 없데이트

공기청정기 가동

T초 후 남은 미세먼지 계산

공기청정기(-1) 부분 제외하고 계산

```
void spread() {
    for(int i=1; i<=R; i++) {
        for(int j=1; j<=C; j++) {
           if (field[i][j] != 0) {
                int subValue = field[i][j] / 5;
                for (int p=0; p<4; p++) {
                    int nx = i + dx[p];
                    int ny = j + dy[p];
                    if (nx < 1 | | nx > R | | ny < 1 | | ny > C) {
                        continue;
                    if (field[nx][ny] == -1) {
                        continue;
                    newField[nx][ny] += subValue;
                    field[i][j] -= subValue;
```

For 문을 돌며 미세먼지 찾기

5로 나눈 몫을 기억해두고,

인접한 상하좌우 필드를 살피기.

범위를 벗어나지 않고, 공기청정기 위치가 아니라면,

해당 필드에 확산된 미세먼지 누적하기.

기존 필드에는 확산된 미세먼지만큼 빼기

```
void afterSpread() {
    for(int i=1; i<=R; i++) {
        for(int j=1; j<=C; j++) {
            field[i][j] += newField[i][j];
            newField[i][j] = 0;
        }
    }
}</pre>
```

확산된 미세먼지 반영하기

다른 필드로부터 확산된 누적 미세먼지 반영하기

→ Field[i][j] += newField[i][j]

시계방향으로 순환

```
void clean() {
    for(int i=airPurifierRoWPos[0]-1; i>1; i--) {
        field[i][1] = field[i-1][1];
    for(int i=1; i<C; i++) {
        field[1][i] = field[1][i+1];
    for(int i=1; i<airPurifierRoWPos[0]; i++) {</pre>
        field[i][C] = field[i+1][C];
    for(int i=C; i>2; i--) {
        field[airPurifierRoWPos[0]][i] =
field[airPurifierRoWPos[0]][i-1];
    field[airPurifierRoWPos[0]][2] = 0;
```

반시계방향으로 순환

바람길을 따라서 한 칸 씩 필드 밀기 복잡해 보이지만 별거 없다..

```
for(int i=airPurifierRoWPos[1]+1; i<R; i++) {</pre>
        field[i][1] = field[i+1][1];
    for(int i=1; i<C; i++) {</pre>
        field[R][i] = field[R][i+1];
    for(int i=R; i>airPurifierRoWPos[1]; i--) {
        field[i][C] = field[i-1][C];
    for(int i=C; i>2; i--) {
        field[airPurifierRoWPos[1]][i] =
field[airPurifierRoWPos[1]][i-1];
    field[airPurifierRoWPos[1]][2] = 0;
```

제출 번호	아이디	문제	결과	메모리	시간	언어	코드 길이
51559107	january	4 17144	맞았습니다!!	2040 KB	40 ms	C++17 / 수정	2730 B
51556768	january	4 17144	틀렸습니다			C++17 / 수정	3596 B

결과