

#파이어볼 명부

- 여기서 사라지면 죽음



ArrayList<FireBall> fireBalls

각 파이어볼의 정보 (행 위치, 열 위치, 질량, 속도, 방향)



FireBall1(행 위치, 열 위치, 질량, 속도, 방향)



FireBall2(행 위치, 열 위치, 질량, 속도, 방향)



FireBall3(행 위치, 열 위치, 질량, 속도, 방향)



FireBall4(행 위치, 열 위치, 질량, 속도, 방향)

>>> [ ,  ,  ,  ,]

#파이어볼 주소록

여기서 사라져도 죽는 것 X



Queue<FireBall>[] map

각 칸마다 이 칸에 포함된 FireBall을 Queue에 담아둠



static void operation0



Queue<FireBall> [][] map

```
for(int i=0;i<N;i++){
    for(int j=0;j<N;j++){
        //합체분리 안해도되는영역
        if(map[i][j].size()<2){
            //애는 이동확인용이라 지움
            map[i][j].clear();// 진짜는 fireBalls임
            continue;
        }
    }
}
```

map 순회

//합체분리해야함

#CASE1. 이번 칸에 파볼 2개 미만 --- $\text{map}[i][j].\text{size}() < 2$

- $\text{map}[i][j]$ 클리어 하고 continue

#CASE2. 이번 칸에 파볼 2개 이상 --- $\text{map}[i][j].\text{size}() \geq 2$

- 합체 및 분리 해야함

1. 새로 만들어질 파볼용 변수 초기화

2. $\text{map}[i][j]$ 에서 poll() 하면서 변수 할당

3. 파볼 생성되나 판단, 생성시 명부에 등록

```
//새로만들어질 파볼 정보
int sumM = 0; //질량합
int sumS = 0; //속도합
int num = curQ.size(); //파볼 수
int odd = 0; // 방향 홀수인 파볼 수
int even = 0; //방향이 짝수인 파볼 수
```

static void operation0

#CASE2. map[i][j] . size0 >= 2

- 합체 및 분리 해야함

1. 새로 만들어질 파볼용 변수 초기화
2. map[i][q]에서 poll() 하면서 변수 할당
3. 파볼 생성되나 판단, 생성시 명부에 등록



Queue<FireBall> [][] map

```
for(int i=0;i<N;i++){
    for(int j=0;j<N;j++){
        //합체분리 안해도되는영역
        if(map[i][j].size()<2){
            //애는 이동확인용이라 지움
            map[i][j].clear();// 진짜는 fireBalls임
            continue;
        }

        //합체분리해야함
```

```
int size = curQ.size();
while(size-->0){//==!curQ.isEmpty()
    //현재 행열에서 파볼하나 뽑기
    FireBall fb = curQ.poll();
    //정보추출
    sumM += fb.m;
    sumS += fb.s;
    num++;이것 때문에 틀림
    if(fb.d%2==0) even++;
    else odd++;
    //합체되므로 재료 제거
    fireBalls.remove(fb);
}
```


static void operation0



```
for(int i=0;i<N;i++){
    for(int j=0;j<N;j++){
        //합체분리 안해도되는영역
        if(map[i][j].size()<2){
            //애는 이동확인용이라 지움
            map[i][j].clear();// 진짜는 fireBalls임
            continue;
        }

        //합체분리해야함
```

#CASE2. map[i][j] . size0 >= 2

- 합체 및 분리 해야함

1. 새로 만들어질 파볼용 변수 초기화
2. map[i][q]에서 poll0 하면서 변수 할당
3. 파볼 생성되나 판단, 생성시 명부에 등록

```
if(sumM/5==0) continue;//질량 0으로 나눠짐
if(odd==0 || even==0){
    fireBalls.add(new FireBall(i,j,sumM/5,sumS/num,0))
    fireBalls.add(new FireBall(i,j,sumM/5,sumS/num,2))
    fireBalls.add(new FireBall(i,j,sumM/5,sumS/num,4))
    fireBalls.add(new FireBall(i,j,sumM/5,sumS/num,6))
}else{
    fireBalls.add(new FireBall(i,j,sumM/5,sumS/num,1))
    fireBalls.add(new FireBall(i,j,sumM/5,sumS/num,3))
    fireBalls.add(new FireBall(i,j,sumM/5,sumS/num,5))
    fireBalls.add(new FireBall(i,j,sumM/5,sumS/num,7))
}
```