```
import numpy as np
import matplotlib.pyplot as plt
N = 100             # number of molecules
D = 100             # side dimension of box
T = 15*10**3        # number of timesteps
B = 50              # location of initial distribution

# initilize variables
nx = 0; ny = 0; s = 0; wallflag = 0
x = np.zeros(N, dtype=int)
y = np.zeros(N, dtype=int)
d = np.zeros(N, dtype=int)
for i in range(N):
    x[i] = np.random.choice(B, 1)
    y[i] = np.random.choice(D, 1)
f = open('workfile', 'w')

for t in range(T):
    R = 0
    for i in range(N):
        if d[i] == 0:
            s = np.random.choice(8, 1); d[i] = s + 1        # random walk 8
directions
        s = d[i]; wallflag = 0                              # inertia
        if s == 1:
            nx = x[i]
            if y[i] + 1 <= D: ny = y[i] + 1; d[i] = s
            else: ny = y[i]; d[i] = 2; wallflag = 1
        if s == 2:
            nx = x[i]
            if y[i] - 1 >= 0: ny = y[i] - 1; d[i] = s
            else: ny = y[i]; d[i] = 1; wallflag = 1
        if s == 3:
            ny = y[i]
            if x[i] + 1 <= D: nx = x[i] + 1; d[i] = s
            else: nx = x[i]; d[i] = 4; wallflag = 1
        if s == 4:
            ny = y[i]
            if x[i] - 1 >= 0: nx = x[i] - 1; d[i] = s
            else: nx = x[i]; d[i] = 3; wallflag = 1
        if s == 5:
            if y[i] + 1 <= D and x[i] + 1 <= D:
                ny = y[i] + 1; nx = x[i] + 1; d[i] = s
            else: ny = y[i]; nx = x[i]; wallflag = 1
            if y[i] + 1 > D: d[i] = 7
            if x[i] + 1 > D: d[i] = 8
        if s == 6:
            if y[i] - 1 >= 0 and x[i] - 1 >= 0:
                ny = y[i] - 1; nx = x[i] - 1; d[i] = s
            else: ny = y[i]; nx = x[i]; wallflag = 1
```

```
            if y[i] - 1 < 0: d[i] = 8
            if x[i] - 1 < 0: d[i] = 7
        if s == 7:
            if y[i] - 1 >= 0 and x[i] + 1 <= D:
                ny = y[i] - 1; nx = x[i] + 1; d[i] = s
            else: ny = y[i]; nx = x[i]; wallflag = 1
            if y[i] - 1 < 0: d[i] = 5
            if x[i] + 1 > D: d[i] = 6
        if s == 8:
            if y[i] + 1 <= D and x[i] - 1 >= 0:
                ny = y[i] + 1; nx = x[i] - 1; d[i] = s
            else: ny = y[i]; nx = x[i]; wallflag = 1
            if y[i] + 1 > D: d[i] = 6
            if x[i] - 1 < 0: d[i] = 5

        if wallflag == 0:
            for j in range(N):
                if x[j] == nx and y[j] == ny and not(j == i):
                    d[i] = 0                                # collision
                x[i] = nx; y[i] = ny                        # inertia

        if x[i] >= int(D/2): R = R + 1

    print(t, R)
    f.write(str(R)+" ")

f.close()
```