Game Probability

Given a rectangular matrix, we can move from the current cell in 4 directions with equal probability. The 4 directions are right, left, top or bottom. Calculate the Probability that after N moves from a given position (i, j) in the matrix, we will not cross boundaries of the matrix at any point.

Input Format:

First line consists of a positive integer, denoting the number of rows of a matrix. 1<=m<=10
Second line consists of a positive integer, denoting number of columns of a matrix 1<=n<=10
Third line denotes the number of steps of movement. N<m

Output Format:

A value between 0 and 1

Solution:

```
def isSafe(x, y, m, n):
    return (x >= 0 and x < m and y >= 0 and y < n)


def findProbability(m, n, x, y, N):




    if (not isSafe(x, y, m, n)):
        return 0.0




    if (N == 0):
```

```
        return 1.0


    prob = 0.0


    prob += findProbability(m, n, x - 1, y, N - 1) * 0.25


    prob += findProbability(m, n, x, y + 1, N - 1) * 0.25


    prob += findProbability(m, n, x + 1, y, N - 1) * 0.25


    prob += findProbability(m, n, x,  y - 1, N - 1) * 0.25


    return prob


m = int(input())
n=int(input())
i = 1
j = 1
N=int(input())
```

```
print(findProbability(m, n, i, j, N))
```