

题目描述：

有一个大小是 $N \times M$ 的战场地图，被墙壁 '#' 分隔成大小不同的区域，上下左右四个方向相邻的空地 '.' 属于同一个区域，只有空地上可能存在敌人 'E'，请求出地图上总共有多少区域里的敌人数小于 K 。

输入描述：

第一行输入为 N, M, K ;

N 表示地图的行数， M 表示地图的列数， K 表示目标敌人数量 $N, M \leq 100$;

之后为一个 $N \times M$ 大小的字符数组。

输出描述：

敌人数小于 K 的区域数量

补充说明：

示例1

输入：3 5 2

..#EE

E.#E.

###..

输出：1

说明：地图被墙壁分为两个区域，左边区域有1个敌人，右边区域有3个敌人，符合条件的区域数量是1

```
n, m, k = [int(_) for _ in input().split()]
```

```
arr = []
```

```
for i in range(n):
```

```
    arr.append(input())
```

```
vis = [[0]*m for _ in range(n)]
```

```
target = 0
```

```
res = 0
```

```
dir = [[0, 1], [1, 0], [0, -1], [-1, 0]]
```

```
def dfs(x, y):
```

```
    if x < 0 or x >= n or y < 0 or y >= m:
```

```
        return
```

```
    if arr[x][y] == '#' or vis[x][y]:
```

```
        return
```

```
    vis[x][y] = 1
```

```
    if arr[x][y] == 'E':
        global target
        target += 1
    for i in range(4):
        dx, dy = dir[i][0]+x, dir[i][1]+y
        dfs(dx, dy)

for i in range(n):
    for j in range(m):
        if arr[i][j] == '#' or vis[i][j]:
            continue
        target = 0
        dfs(i, j)
        if target < k:
            res += 1

print(res)
```