

### 机器人运动范围

地上有一个  $m$  行  $n$  列的方格，从坐标  $[0,0]$  到坐标  $[m-1,n-1]$ 。一个机器人从坐标  $[0,0]$  的格子开始移动，它每次可以向左、右、上、下移动一格（不能移动到方格外），也不能进入行坐标和列坐标的数位之和大于  $k$  的格子。例如，当  $k$  为 18 时，机器人能够进入方格  $[35,37]$ ，因为  $3+5+3+7=18$ 。但它不能进入方格  $[35,38]$ ，因为  $3+5+3+8=19$ 。请问该机器人能够到达多少个格子？

示例：

输入：  $m = 2, n = 3, k = 1$

输出： 3

```
class Solution {
public:
    int movingCount(int m, int n, int k) {
        if(k == 0) return 1;
        vector<vector<bool>> valid(m, vector<bool>(n, true)); // 记录该位置
        // 是否被访问过
        return dfs(valid, m, n, 0, 0, k);
    }
    int dfs(vector<vector<bool>> & valid, int m, int n, int row, int col, i
nt k) // valid 的传值一定要用 &
    {
        int sum = getSum(row) + getSum(col);
        // 如果越界，或者和大于 k，或者已被访问过了，返回 0
        if(row >= m || col >= n || sum > k || !valid[row][col]) return 0;
        valid[row][col] = false; // 该位置状态变为：已访问过
        return 1 + dfs(valid, m, n, row+1, col, k) + dfs(valid, m, n, row, col+1, k);
        // 回溯法（递归）
    }
    int getSum(int num)
    {
        // 求某个数字所有位数相加的和
        if(num < 10) return num;
        int sum = 0;
        while(num > 0)
        {
            sum += num % 10;
            num /= 10;
        }
        return sum;
    }
};
```