



链滴

机器人的运动范围

作者: [yudake](#)

原文链接: <https://ld246.com/article/1519091177881>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

题目描述

地上有一个m行和n列的方格。一个机器人从坐标0,0的格子开始移动，每一次只能向左，右，上，下四个方向移动一格，但是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 $3+5+3+7=18$ 。但是，它不能进入方格(35,38)，因为 $3+5+3+8=19$ 。请问该机器人能够达到多少个格子？

解题思路

回溯法。利用标志位记录已经走过的位置，不过这里不需要对其去标志处理，任何一种方法到达都可。然后记录在ret矩阵中，或者计算标志位中标志的个数。

代码

```
public class Solution {
    public int movingCount(int threshold, int rows, int cols) {
        int[] ret = new int[1];
        boolean[][] label = new boolean[rows][cols];
        canThis(threshold, rows, cols, 0, 0, label, ret);
        return ret[0];
    }

    private void canThis(int threshold, int rows, int cols, int row, int col, boolean[][] label, int[] ret) {
        if (row < 0 || col < 0 || row >= rows || col >= cols || label[row][col] == true)
            return;
        int k = 0;
        int tcol = col;
        int trow = row;
        while (trow != 0) {
            k += trow % 10;
            trow /= 10;
        }
        while (tcol != 0) {
            k += tcol % 10;
            tcol /= 10;
        }
        if (k <= threshold) {
            ret[0]++;
            label[row][col] = true;
            canThis(threshold, rows, cols, row+1, col, label, ret);
            canThis(threshold, rows, cols, row-1, col, label, ret);
            canThis(threshold, rows, cols, row, col+1, label, ret);
            canThis(threshold, rows, cols, row, col-1, label, ret);
        } else
            return;
    }
}
```