[编程题]二维数组中的查找

在一个二维数组中（每个一维数组的长度相同），每一行都按照从左到右递增的顺序排序，每一列都按照从上到下递增的顺序排序。请完成一个函数，输入这样的一个二维数组和一个整数，判断数组中是否含有该整数

思路：就像一个扇形，距离圆心的比半径小则在扇形内，等于在扇形上，大于在扇形外

/\* 思路

\* 矩阵是有序的，从左下角来看，向上数字递减，向右数字递增，

\* 因此从左下角开始查找，当要查找数字比左下角数字大时。右移

\* 要查找数字比左下角数字小时，上移

\*/

答案：

class Solution {

public:  
    bool Find(vector<vector<int> > array,int target) {  
        int rowCount = array.size();  
        int colCount = array[0].size();  
        int i,j;  
        for(i=rowCount-1,j=0;i>=0&&j<colCount;)  
        {  
            if(target == array[i][j])  
                return true;  
            if(target < array[i][j])  
            {  
                i--;  
                continue;  
            }  
            if(target > array[i][j])  
            {  
                j++;  
                continue;  
            }  
        }  
        return false;  
    }  
};