

## 牛客网-华为机试练习题 92

### 题目描述

请编写一个函数（允许增加子函数），计算 $n \times m$ 的棋盘格子（ $n$ 为横向的格子数， $m$ 为竖向的格子数）沿着各自边缘线从左上角走到右下角，总共多少种走法，要求不能走回头路，即：只能往右和往下走，不能往左和往上走。

### 输入描述:

输入两个正整数

### 输出描述:

返回结果

示例1

输入

2

2

输出

6

### 解决代码:

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
/* 横向n个格子，纵向 m个， n+1,m+1个交点
 * 只能往右和下走
 * dp[n+1][m+1] 每一次向下走和向右走将会导致完全不一样的路线，并且只能向下或向右走
 * dp[i][j]=dp[i][j-1]+dp[i-1][j]
 */
public class Main {
    public static int getCount(int n,int m) {
        int[][] dp=new int[n+1][m+1];
        for(int i=0;i<n+1;i++) {
            for(int j=0;j<m+1;j++) {
                if(i==0||j==0) dp[i][j]=1;
                else dp[i][j]=dp[i][j-1]+dp[i-1][j];
            }
        }
        return dp[n][m];
    }
    public static void main(String[] args)throws IOException{
        BufferedReader bReader = new BufferedReader(new InputStreamReader(System.in));
        String line=null;
        while((line=bReader.readLine())!=null) {
            int n=Integer.valueOf(line.substring(0,line.indexOf(" ")));
            int m=Integer.valueOf(line.substring(line.indexOf(" ")+1));
            System.out.println(getCount(n, m));
        }
    }
}
```

