## 2041 超级楼梯

### 一、题目

#### 问题描述

有一楼梯共M级，刚开始时你在第一级，若每次只能跨上一级或二级，要走上第M级，共有多少种走法？

#### 输入数据

输入数据首先包含一个整数N，表示测试实例的个数，然后是N行数据，每行包含一个整数M（1<=M<=40）,表示楼梯的级数。

#### 输出数据

对于每个测试实例，请输出不同走法的数量。

#### 输入样例

2 2 3

#### 输出样例

1 2

#### 题目来源

HDU 2041 http://acm.hdu.edu.cn/showproblem.php?pid=2041

### 二、题解

#### 解题思路

这道题蕴含了动态规划的思想，可以从最后一层考虑，当有M层楼梯时，如果最后一层不走，则前按第M-1层的走法再走一步 如果最后留下两层不走，则是按第M-2层的走法再走一步 总的方法就是前两层方法的和。其实质是一个斐波那契数列。可以考虑使用循环的方式实现，也可以使用递归实现

#### 参考程序

#include<stdio.h>  
int step(int n)//求阶梯的走法  
{  
 if(n==1)  
 return 1;  
 else if(n==2)  
 return 2;  
 else   
 return step(n-1)+step(n-2);  
}  
int main()  
{  
 int M,N;  
 int i,sum;  
 while(scanf("%d",&N)!=EOF){  
 while(N--){  
 sum=0;  
 scanf("%d",&M);  
 M=M-1;//因为从第一层开始，所以减一  
 sum=step(M);  
 printf("%d\n",sum);  
 }  
 }  
 return 0;  
}

#### 复杂度分析

无

#### 编程技巧

无