

第五章 数的存储与组织

第 6 7 课 递推

《信息学奥赛一本通·编程启蒙 C++版》

【例 67.1】 爬楼梯

【题目描述】

树老师爬楼梯，他可以每次走 1 级或者 2 级，输入楼梯的级数，求不同的走法数。

例如：楼梯一共有 3 级，他可以每次都走一级，或者第一次走一级，第二次走两级，也可以第一次走两级，第二次走一级，一共 3 种方法。

【输入格式】

输入包含若干行，每行包含一个正整数 N ，代表楼梯级数， $1 \leq N \leq 30$ 。

【输出格式】

不同的走法数，每一行输入对应一行输出。

【样例输入】

5

8

10

【样例输出】

8

34

89

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int x;
4. int a[35];
5. int main(){
6.     while(cin>>x){
7.         a[1]=1;
8.         a[2]=2;
9.         for(int i=3;i<=x;i++){
10.            a[i]=a[i-1]+a[i-2];
11.        }
12.        cout<<a[x]<<endl;
13.    }
14.    return 0;
15.}
```

【例 67.2】 昆虫繁殖

【题目描述】

科学家在热带森林中发现了一种特殊的昆虫，这种昆虫的繁殖能力很强。每对成虫过 x 个月产 y 对卵，每对卵要过两个月长成成虫。假设每个成虫不死，第一个月只有一对成虫，且卵长成成虫后的第一个月不产卵(过 x 个月产卵)，问过 z 个月以后共有成虫多少对？

$$1 \leq x \leq 20, 1 \leq y \leq 20, x < z \leq 50。$$

【输入格式】

x, y, z 的数值。

【输出格式】

过 z 个月以后，共有成虫对数。

【样例输入】

1 2 8

【样例输出】

37

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int x,y,n;
4. long long bug[105],egg[105];
5. long long deal(int x){
6.     long long sum=0;
7.     for(int i=x;i>=1;i--){
8.         sum+=bug[i];
9.     }
10.    return sum;
11.}
12.int main(){
13.    cin>>x>>y>>n;
14.    bug[1]=1;
15.    for(int i=2;i<=n+1;i++){
16.        egg[i]=deal(i-x)*y;
17.        bug[i]=egg[i-2];
18.    }
19.    cout<<deal(n+1);
20.    return 0;
21.}
```

【例 67.3】 数字金字塔

【题目描述】

观察下面的数字金字塔。写一个程序查找从最高点到底部任意处结束的路径，使路径经过数字的和最大。每一步可以从当前点走到左下方的点也可以到达右下方的点：



$1 \leq x \leq 20, 1 \leq y \leq 20, x < z \leq 50$ 。

【输入格式】

第一个行包含 **R** ($1 \leq R \leq 1000$)，表示行的数目。后面每行为这个数字金字塔特定行包含的整数。

所有的被供应的整数是非负的且不大于 **100**。。

【输出格式】

单独的一行，包含那个可能得到的最大的和。

【样例输入】

```

5
13
11 8
12 7 26
6 14 15 8
12 7 13 24 11

```

【样例输出】

86

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3. int main(){
4.     int n,i,j,a[1001][1001];
5.     cin>>n;
6.     for(i=1;i<=n;i++)
7.         for(j=1;j<=i;j++)
8.             cin>>a[i][j];
9.     for(i=n-1;i>=1;i--)
10.        for(j=1;j<=i;j++)
11.            if(a[i+1][j]>=a[i+1][j+1])
12.                a[i][j]+=a[i+1][j];
13.            else a[i][j]+=a[i+1][j+1];
14.     cout<<a[1][1];
15.     return 0;
16.}
```


练 67.1 斐波那契数列

【题目描述】

斐波那契数列是指这样的数列:数列的第一个和第二个数都为 1, 接下来每个数都等于前面 2 个数之和。给出一个正整数 a, 要求斐波那契数列中第 a 个数对 1000 取模的结果是多少。

【输入格式】

第 1 行是测试数据的组数 n, 后面跟着 n 行输入。每组测试数据占 1 行, 包括一个正整数 a ($1 \leq a \leq 1000000$)

【输出格式】

n 行, 每行输出对应一个输入。输出应是一个正整数, 为斐波那契数列中第 a 个数对 1000 取模得到的结果。

【样例输入】

4
5
2
19
1

【样例输出】

5
1
181
1

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3. int n;
4. int a[1000005],b[1000005];
5. int main(){
6.     cin>>n;
7.     for(int i=1;i<=n;i++){
8.         cin>>a[i];
9.     }
10.    b[1]=1;
11.    for(int i=2;i<=1000000;i++){
12.        b[i]=(b[i-1]+b[i-2])%1000;
13.    }
14.    for(int i=1;i<=n;i++){
15.        cout<<b[a[i]]<<endl;
16.    }
17.    return 0;
18.}
```

练 67.2 金猪大家庭

【题目描述】

金猪大家庭主要由母猪构成的，小母猪第二年就可以生 2 只小母猪和 2 只小公猪，第三年开始每年可以生 4 只小母猪和 4 只小公猪；小公猪在第二年就要被赶出大家庭；猪的寿命只有六年。

目前，一个新的金猪大家庭正在形成：一只小母猪自立门户，创建自己的金猪大家庭。

【输入格式】

只有一行且只有一个正整数： $n (1 \leq n \leq 35)$

【输出格式】

只有一行且只有一个正整数：第 n 年金猪大家庭的成员数。

【样例输入】

3

【样例输出】

19

【代码实现】

```
1. #include<bits/stdc++.h>
2. using namespace std;
3. int main(){
4.     long long s,a[50]={0};
5.     int n,i;
6.     cin>>n;
7.     if(n==1){
8.         cout<<1<<endl;
9.     }
10.    else{
11.        n+=5;
12.        s=0;
13.        a[6]=1;
14.        for(i=7;i<=n;i++){
15.            s+=a[i-2]-a[i-6];
16.            a[i]=a[i-1]*2+s*4;
17.        }
18.        cout<<s+a[n-1]+2*a[n]<<endl;
19.    }
20.    return 0;
21.}
```

练 67.3 背单词

【题目描述】

SLM 小朋友为了准备英语考试，发了疯地背单词。SLM 小朋友决定每天背一次单词，正常情况下，他会在起床前大约 6:20-7:00 背单词。并且他有一个技能，就是在第 i 天的 23:35 开始背单词，一直背到第 $i+1$ 天的 0:25 分，这样就算两天各背了一次单词。但是这个技能不能连续使用两次，否则会累死。比如：第一天和第二天用技能，第三天和第四天也用技能，这是不行的。但是第一天和第二天用技能，第三天正常背，第四天和第五天用技能这是可以的。

SLM 小朋友想要背 n 天单词，他想知道有多少种合法的背单词方式。注意，如果在第 n 天和第 $n+1$ 天使用技能或者在第 0 天和第一天使用技能，这是不被允许的。

【输入格式】

一个正整数 n , $n \leq 80$

【输出格式】

一个整数，即背单词的方案数。

【样例输入】

5

【样例输出】

6

【代码实现】

```
1. #include<iostream>
2. using namespace std;
3. int main()
4. {
5.     long long n,a[100];
6.     a[1]=1;
7.     a[2]=2;
8.     a[3]=3;
9.     cin>>n;
10.    for(int i=4;i<=n;i++)
11.        a[i]=a[i-1]+a[i-3];
12.    cout<<a[n]<<' \n';
13.}
```

谢谢！

