POJ-3070Fibonacci（矩阵快速幂求Fibonacci数列）

Fibonacci

Time Limit: 1000MS		**Memory Limit:** 65536K
Total Submissions: 7241		**Accepted:** 5131

Description

In the Fibonacci integer sequence, $F_0 = 0$, $F_1 = 1$, and $F_n = F_{n-1} + F_{n-2}$ for $n \geq 2$. For example, the first ten terms of the Fibonacci sequence are:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, …

An alternative formula for the Fibonacci sequence is

$$\begin{bmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^n = \underbrace{\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \cdots \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}}_{n \text{ times}}$$

.

Given an integer n, your goal is to compute the last 4 digits of F_n.

Input

The input test file will contain multiple test cases. Each test case consists of a single line containing n (where $0 \leq n \leq 1{,}000{,}000{,}000$). The end-of-file is denoted by a single line containing the number −1.

Output

For each test case, print the last four digits of F_n. If the last four digits of F_n are all zeros, print '0'; otherwise, omit any leading zeros (i.e., print F_n mod 10000).

Sample Input

```
0
9
999999999
1000000000
-1
```

Sample Output

```
0
34
```

昵称：可笑痴狂
园龄：4年9个月
粉丝：154
关注：35
+加关注

< 2013年6月 >

日	一	二	三	四	五	六
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

搜索

找找看

谷歌搜索

最新随笔

1. 批处理实现批量创建快捷方式
2. 设计模式（六）装饰模式（转）
3. 设计模式（五）桥接模式（转）
4. 设计模式（四）适配器模式（转）

626
6875

Hint

As a reminder, matrix multiplication is associative, and the product of two 2 × 2 matrices is given by

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} & a_{11}b_{12} + a_{12}b_{22} \\ a_{21}b_{11} + a_{22}b_{21} & a_{21}b_{12} + a_{22}b_{22} \end{bmatrix}$$

.

Also, note that raising any 2 × 2 matrix to the 0th power gives the identity matrix:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}^0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Source

Stanford Local 2006

分析：通过这道题，不仅学会了矩阵的快速幂的做法，同时也提供了求Fibonacci的高效算法

代码一：

这题完全套用的是 一般的快速幂的做法，只不过改成矩阵乘法后，为了在做矩阵乘法过程中不会影响结果值，
之间要用中间变量，代码写的很难看 （没有想到可以用结构体对二维数组进行封装，可以直接返回结构体类型的数据），
不过还是 0ms AC。

```
#include <cstdio>
#include <iostream>

using namespace std;

const int MOD = 10000;

int fast_mod(int n)     // 求 (t^n)%MOD
{
    int t[2][2] = {1, 1, 1, 0};
    int ans[2][2] = {1, 0, 0, 1};   // 初始化为单位矩阵
    int tmp[2][2];     //自始至终都作为矩阵乘法中的中间变量

    while(n)
    {
        if(n & 1)  //实现 ans *= t; 其中要先把 ans赋值给 tmp，
```


随笔分类(487)

随笔档案(390)

```
然后用 ans = tmp * t
17          {
18              for(int i = 0; i < 2; ++i)
19                  for(int j = 0; j < 2; ++j)
20                      tmp[i][j] = ans[i][j];
21              ans[0][0] = ans[1][1] = ans[0][1] = ans[1][0]
= 0;  // 注意这里要都赋值成 0
22
23              for(int i = 0; i < 2; ++i)     //  矩阵乘法
24              {
25                  for(int j = 0; j < 2; ++j)
26                  {
27                      for(int k = 0; k < 2; ++k)
28                          ans[i][j] = (ans[i][j] + tmp[i][k]
* t[k][j]) % MOD;
29                  }
30              }
31          }
32
33          //  下边要实现  t *= t 的操作，同样要先将t赋值给中间变量
tmp ，t清零，之后 t = tmp* tmp
34          for(int i = 0; i < 2; ++i)
35              for(int j = 0; j < 2; ++j)
36                  tmp[i][j] = t[i][j];
37          t[0][0] = t[1][1] = 0;
38          t[0][1] = t[1][0] = 0;
39          for(int i = 0; i < 2; ++i)
40          {
41              for(int j = 0; j < 2; ++j)
42              {
43                  for(int k = 0; k < 2; ++k)
44                      t[i][j] = (t[i][j] + tmp[i][k] *
tmp[k][j]) % MOD;
45              }
46          }
47
48          n >>= 1;
49      }
50      return ans[0][1];
51  }
52
53  int main()
54  {
55      int n;
56      while(scanf("%d", &n) && n != -1)
57      {
58          printf("%d\n", fast_mod(n));
59      }
60      return 0;
61  }
```

文章分类(34)

文章档案(34)

代码二：用结构体封装矩阵乘法后，代码看着清晰多了

```
#include <cstdio>
#include <iostream>

using namespace std;

const int MOD = 10000;

struct matrix
{
    int m[2][2];
}ans, base;

matrix multi(matrix a, matrix b)
{
    matrix tmp;
    for(int i = 0; i < 2; ++i)
    {
        for(int j = 0; j < 2; ++j)
        {
            tmp.m[i][j] = 0;
            for(int k = 0; k < 2; ++k)
                tmp.m[i][j] = (tmp.m[i][j] + a.m[i][k] * b.m[k][j]) % MOD;
        }
    }
    return tmp;
}
int fast_mod(int n)  // 求矩阵 base 的  n 次幂
{
    base.m[0][0] = base.m[0][1] = base.m[1][0] = 1;
    base.m[1][1] = 0;
    ans.m[0][0] = ans.m[1][1] = 1;  // ans 初始化为单位矩阵
    ans.m[0][1] = ans.m[1][0] = 0;
    while(n)
    {
        if(n & 1)  //实现 ans *= t; 其中要先把 ans赋值给 tmp，然后用 ans = tmp * t
        {
            ans = multi(ans, base);
        }
        base = multi(base, base);
        n >>= 1;
    }
    return ans.m[0][1];
}

int main()
{
    int n;
```

2013年7月 (1)
2013年3月 (1)
2012年12月 (2)
2012年11月 (11)
2012年10月 (2)
2012年9月 (1)
2012年8月 (12)
2012年7月 (2)
2012年6月 (1)

博客链接

开开甲
三江小渡
算法导论学习博客
算法天才
算法专题
先前新浪博客
知行

积分与排名

积分 - 257042
排名 - 609

最新评论

1. Re:NYOJ-90整数划分
解释得真清楚，谢谢
--SherylHan

2. Re:NYOJ-214 单调递增子序列(二)
(2) A[x] A[x], F[y]不是应该大于F[x]吗？？...
--ThompsonLin

3. Re:C++抽象类
楼主两处构造函数没有加花括号，报错：undefined reference to。。。
--lawlietfans

4. Re:Windows共享内存示例
简单粗暴，比网上讲一大堆的管用；我说呢，linux下共享内存那么简单咋在windows上会那么麻烦；谢谢啦，另外string转LPCWSTR需要写个函数：LPCWSTR stringToLPCWST.....
--北辰_Yx

5. Re:子集生成算法
两个都是错的
--墨城烟火

6. Re:第一个手写Win32窗口程序
aghfdhdhdgfdghuirwhtuigrg

```
48      while(scanf("%d", &n) && n != -1)
49      {
50          printf("%d\n", fast_mod(n));
51      }
52      return 0;
53  }
```

功不成，身已退

分类: 数论+计算几何

标签: POJ-3070Fibonacci（矩阵快速幂求Fibonacci数列）

好文要顶 关注我 收藏该文

可笑痴狂
关注 - 35
粉丝 - 154
+加关注

2 0

« 上一篇：NYOJ-172 小珂的图表
» 下一篇：HDOJ-1058 Humble Numbers

posted on 2013-06-02 17:08 可笑痴狂 阅读(9541) 评论(1) 编辑 收藏

FeedBack:

#1楼 2015-05-15 16:32 | liuke0002

boom!~

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 登录 或 注册，访问网站首页。

最新IT新闻:
· 2017年软件开发人员需要面对的7个改变
· 腾讯手游业务首次单季度营收超百亿，鹅厂有王者荣耀就够了
· 苏宁云商称已出售PPTV股权至公司实控人张近东的企业
· 微软云服务又宕机了，这是本月第2次
· 腾讯市值1.86万亿元 力压阿里成市值最高新兴市场公司
» 更多新闻...

最新知识库文章:
· 程序员学习能力提升三要素
· 为什么我要写自己的框架？
· 垃圾回收原来是这么回事
· 「代码家」的学习过程和学习经验分享
· 写给未来的程序媛
» 更多知识库文章...

--312869788
7. Re:第一个手写Win32窗口程序
说过热按规范噶电话的就会发生更热爱第三方公司的
--312869788
8. Re:HDOJ-2553 N皇后问题
--lvable
9. Re:HDOJ-3790 最短路径问题
偶错了，dis[]和cost1[]局部数组还是需要的。。。
--banana16314
10. Re:HDOJ-3790 最短路径问题
但是，偶觉得dis[],cost1[]数组不需要啊。。。
--banana16314

阅读排行榜

1. Oracle中Merge into用法总结(47822)
2. C++抽象类(18598)
3. table表单中的属性(17180)
4. 在VS中添加lib库的三种方法(15072)
5. POJ-3070Fibonacci（矩阵快速幂求Fibonacci数列）(9541)

评论排行榜

1. HDU 1869 六度分离 -----Floyd(5)
2. NYOJ-68 三点顺序 --有向面积(4)
3. HDOJ-3790 最短路径问题(3)
4. Mysql常用命令大全(2)
5. poj-2080 Calendar(2)

推荐排行榜

1. Mysql常用命令大全(5)
2. POJ-1664 放苹果(5)
3. Oracle中Merge into用法总结(3)
4. 在VS中添加lib库的三种方法(3)
5. multiMap遍历方法(2)

Copyright ©2017 可笑痴狂 Powered By博客园 模板提供：沪江博客