

导航

- 博客园
- 首页
- 新随笔
- 联系
- 订阅 
- 管理

< 2013年3月 >						
日	一	二	三	四	五	六
24	25	26	27	28	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
31	1	2	3	4	5	6

公告

昵称：旭东的博客
园龄：4年4个月
粉丝：232
关注：12
+加关注

搜索

找找看

谷歌搜索

随笔分类

- C/C++学习(36)
- GPU/CUDA(3)
- GPU/OpenCL(8)
- Html/JS(9)
- java编程(28)
- Linux下C编程(10)
- Linux学习(20)
- OpenMP(3)
- PHP(4)
- Python(5)
- Shell(4)
- SQL(6)
- SSH框架学习(2)
- 概念(7)
- 模式匹配(3)
- 其他(2)
- 设计模式(5)
- 算法学习(27)
- 网页爬虫(2)
- 文本压缩(4)
- 总结汇总(4)

随笔档案

- 2017年3月 (4)
- 2017年2月 (1)
- 2016年11月 (1)
- 2016年5月 (2)
- 2016年3月 (1)

斐波那契数列 矩阵求法 优化

在做编程题目的时候经常会遇到“斐波那契数列”相关的题目，尤其在做OJ中。下面说一些方法：

(一)递归

递归是最慢的会发生重复计算，时间复杂度成指数级。



```
long long fac(int n)
{
    if (n==1) return 1;
    else if (n==2) return 2;
    else return fac(n-1)+fac(n-2);
}
```



(二)循环

利用临时变量来保存中间的计算过程，加快运算。



```
long long fac(int n)
{
    long long a=1,b=2,c;
    if (n==1) return 1;
    else if (n==2) return 2;
    else
    {
        for (int i=3;i<=n;i++)
        {
            c=a+b; a=b; b=c;
        }
    }
    return b;
}
```



(三)矩阵乘法+空间换时间(减少乘法，取模运算)

数列的递推公式为：f(1)=1，f(2)=2，f(n)=f(n-1)+f(n-2)(n>=3)

用矩阵表示为：

$$\begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} f(n-1) \\ f(n-2) \end{bmatrix}$$

进一步，可以得出直接推导公式：

2015年9月 (1)
 2014年11月 (1)
 2014年10月 (7)
 2014年9月 (8)
 2014年8月 (5)
 2014年7月 (3)
 2014年6月 (8)
 2014年5月 (10)
 2014年4月 (10)
 2014年3月 (17)
 2014年2月 (5)
 2014年1月 (8)
 2013年12月 (15)
 2013年11月 (7)
 2013年10月 (2)
 2013年9月 (3)
 2013年7月 (7)
 2013年6月 (1)
 2013年5月 (11)
 2013年4月 (12)
 2013年3月 (14)
 2013年1月 (2)

文章分类

Shell

阅读排行榜

1. Java 读写Properties配置文件(114014)
2. [转]linux awk命令详解(78065)
3. Java Arrays类进行数组排序(55561)
4. Linux C Socket编程原理及简单实例(44255)
5. Html form 表单提交前验证(43459)

推荐排行榜

1. Java 读写Properties配置文件(11)
2. C++ const用法 尽可能使用const(9)
3. Linux C Socket编程原理及简单实例(9)
4. java中String、StringBuffer、StringBuilder的区别(9)
5. java中的==、equals()、hashCode()源码分析(7)

$$\begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} f(n-1) \\ f(n-2) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^2 \begin{bmatrix} f(n-2) \\ f(n-3) \end{bmatrix} = \dots = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{n-2} \begin{bmatrix} f(2) \\ f(1) \end{bmatrix}$$

由于**矩阵乘法满足结合律**，在程序中可以事先给定矩阵的64，32，16，8，4，2，1次方，**加快程序的执行时间**。（有些题目需要取模运算，也可以事先进行一下）。给定的矩阵次幂，与二进制有关是因为，如下的公式**存在解**，满足 $X_i = \{0 \text{ 或 } 1\}$ ：

$$x_0 * 2^0 + x_1 * 2^1 + x_2 * 2^2 + \dots + x_n * 2^n = A$$

为了保证解满足 $X_i = \{0 \text{ 或 } 1\}$ ，对上述公式的求解从右向左，即求解顺序为 $X_n, X_{n-1}, X_{n-2}, \dots, X_1, X_0$ 。

完整代码实现如下：



```

//求解fac(n)%100000，其中n为大于等于3的正整数
#include<stdio.h>
#include<math.h>
long long fac_tmp[6][4]={    //存放矩阵次幂
    //位置：00 01 10 11
    {24578,78309,78309,46269},    //32次幂%100000
    {1597,987,987,610},    //16次幂%100000
    {34,21,21,13},    //8次幂%100000
    {5,3,3,2},    //4次幂%100000
    {2,1,1,1},    //2次幂%100000
    {1,1,1,0},    //1次幂%100000
};

void fac(int);

int main()
{
    int n;
    scanf("%d",&n);
    fac(n);
    return 1;
}

void fac(int k) //k>=3
{
    int i;
    long long t00=1,t01=1,t10=1,t11=0;    //表示矩阵的1次幂
    long long a,b,c,d;
    k=k-3;    //公式中是n-2次幂，(t00,t01,t10,t11)表示1次幂。所以一共减3次
    for(i=k;i>=32;i=i-32)    //对于大于等于32的k;
    {
        a=(t00*fac_tmp[0][0]+t01*fac_tmp[0][2])%100000;
        b=(t00*fac_tmp[0][1]+t01*fac_tmp[0][3])%100000;
        c=(t10*fac_tmp[0][0]+t11*fac_tmp[0][2])%100000;
        d=(t10*fac_tmp[0][1]+t11*fac_tmp[0][3])%100000;
        t00=a;    t01=b;    t10=c;t11=d;
    }
}
  
```

```
i=4;
while(i>=0)    ///对于小于32的k(16,8,4,2,1);
{
    if(k>=(long long)pow(2,i))    ///如果k大于某一个2的次幂
    {
        a=(t00*fac_tmp[5-i][0]+t01*fac_tmp[5-i][2])%100000;
        /// (5-i) : 矩阵的2的i次幂在数组fac_tmp中的位置为fac_tmp[5-i]
        b=(t00*fac_tmp[5-i][1]+t01*fac_tmp[5-i][3])%100000;
        c=(t10*fac_tmp[5-i][0]+t11*fac_tmp[5-i][2])%100000;
        d=(t10*fac_tmp[5-i][1]+t11*fac_tmp[5-i][3])%100000;
        t00=a;  t01=b;  t10=c; t11=d;
        k=k-(int)pow(2,i);
    }
    i--;
}

a=(t00*2+t01*1)%100000;
printf("%lld\n",a);
}
```

分类: [算法学习](#)

好文要顶

关注我

收藏该文



旭东的博客

关注 - 12

粉丝 - 232

2

0

[+加关注](#)« 上一篇: [物件捆绑 背包问题 动态规划 求解](#)» 下一篇: [Java数据导入 \(读 \) Excel文件 解析](#)

posted on 2013-03-19 10:39 旭东的博客 阅读(9206) 评论(2) 编辑 收藏

评论

#1楼

。给定的矩阵次幂，与二进制有关是因为，如下的公式存在解，满足 $X_i = \{0 \text{ 或 } 1\}$ ：

这一段没看懂，后续代码也没看出什么关系

支持(0) 反对(0)

2015-04-25 10:57 | haithink

#2楼

看出点关系了。

是这个意思吧：

把任意一个数字转换成 2 的幂次方的和

支持(0) 反对(0)

2015-04-25 10:58 | haithink

[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：

- 机器人也玩足球 优必选Alpha2成为曼城2号球员
 - 腾讯、阿里双双宣布使用英特尔Optane SSD
 - 京东因涉嫌虚假宣传被起诉 法院已受理
 - 滴滴确认将停止对非京牌车主派单：按新政要求逐步过渡
 - 给“共享单车”算一笔账，未来18个月见生死
- » [更多新闻...](#)

最新知识库文章：

- 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
 - 高质量的工程代码为什么难写
- » [更多知识库文章...](#)

Powered by:

博客园

Copyright © 旭东的博客