数学方法专题

【主要内容】

一、公约数
二、素数
三、置换
四、解递推方程的几个实例

【试题分析与讲解】

【例 1】公约数

a,b 是两个不全为 0 的整数，他们的最大公约数记作 gcd(a,b)，最小公倍数记作 lcm(a,b)
命题：a*b=gcd(a,b)*lcm(a,b)

证明：$a=P_1^{a_1}*P_2^{a_2}\cdots\cdots*P_m^{a_m}$

$$b=P_1^{b_1}*P_2^{b_2}\cdots\cdots*P_m^{b_m}$$

$$\mathrm{gcd}(\mathrm{a},\mathrm{b})=P_1^{\min(a_1,b_1)}*P_2^{\min(a_2,b_2)}\cdots\cdots*P_m^{\min(a_m,b_m)}$$

$$\mathrm{lcm}(\mathrm{a},\mathrm{b})=P_1^{\max(a_1,b_1)}*P_2^{\max(a_2,b_2)}\cdots\cdots*P_m^{\max(a_m,b_m)}$$

而 $\max(\mathrm{a_i},\mathrm{b_i})+\min(\mathrm{a_i},\mathrm{b_i})=\mathrm{a_i}+\mathrm{b_i}$

$$\mathrm{a}*\mathrm{b}=P_1^{a_1+b_1}*P_2^{a_2+b_2}\cdots\cdots*P_m^{a_m+b_m}$$

$$=P_1^{\min(a_1,b_1)+\max(a_1,b_1)}*P_2^{\min(a_2,b_2)+\max(a_2,b_2)}\cdots\cdots*P_m^{\min(a_m,b_m)+\max(a_m,b_m)}$$

=gcd(a,b)*lcm(a,b)

命题成立

【例 2】欧几里德辗转相除法

定理：gcd(a,b)=gcd(b,a mod b)

PASCAL 程序段：

```
Function gcd(a, b :integer):integer;
    begin
        if  b=1 then gcd:=a
                        else  gcd:=gcd(b, a mod b)
    end;
```

C++程序段：

```
int gcd(int a, int b)
{
      return (b=1)?a:(gcd(b, a%b));
```

}

【例 3】扩展欧几里德算法

若存在一组解 x_0, y_0，满足

$b*x_0$+(a mod b)*y_0=d

则取 x=y_0， y=x_0-(a div b)*y_0，有

ax+by=d

这样我们可以用类似辗转相除的迭代法求解。

程序：

```
Function extended-gcd(a, b:longint; var x, y:integer);
Var  x1, y1 :integer;
Begin
      if b=0 then begin
            extended-gcd:=a;
            x:=1; y:=0
      end else begin
            extended-gcd:=extended-gcd(b, a mod b, x1, y1);
            x:=y1;
            y:= x1 -(a div b)* y1 ;
      end;
  end;
```

【例 4】素数

大于 1 的正整数 p 被称作素数，如果 p 仅有的正因子是 1 和 p

素数判定

理论：如果 n 是合数，则他必有一个小于或等于 sqrt(n)的约数。

操作：枚举 2 至[sqrt(n)]的所有整数，判断其中有无 p 的约数。

时间复杂度：O(sqrt(n))

求 2-n 中所有的素数：

筛选法：

建立一个表，给每个值标记 true

从 2 到 n 枚举每一个整数 p，如果当前 p 的标记为 true，则 p 为素数

若当前 p 为素数，则将 n 以内的 p*p , p*(p+1),…都标记为 false（被筛去）

【例 5】Fibonacci 素数

Fibonacci 数列：1,1,2,3,5,8,…

若某一个 Fibonacci 数与任何比它小的 Fibonacci 数互质，则将其称为 Fibonacci 素数。要求输出第 k 小的 Fibonacci 素数。

考察所有 Fibonacci 数对 M 取余的序列{ S_i }

$S_1 = S_2 =1$

设第一个零元素为 S_k =0，那么必有 $S_{k+1} = S_{k+2}$ =a，故从 k+1 项开始，相当于前面的序列乘以 a。故当且仅当 p 为 k 的倍数时，S_p =0

因此对于 a>b>=1，如果 a 是 b 的倍数，则 f_a 是 f_b 的倍数.

结论：

第 1，2 项为 1，不是 Fibonacci 素数

第 4 项也是 Fibonacci 素数，因为 4 虽然是 2 的倍数，但 F_2 =1

其他的项中，某项为 Fibonacci 素数，当且仅当它的项数为素数。即 3,5,7,11,13…项为 Fibonacci 素数。

【例 6】置换

n 个元素 1,2,…,*n* 之间的一个置换

1， 2， 3， 4，……，n

a_1， a_2， a_3， a_4 ……， a_n

表示 1 被 1 到 *n* 中的某个数 a_1 取代，2 被 1 到 *n* 中的某个数 a_2 取代，直到 *n* 被 1 到 *n* 中的某个数 a_n 取代，且 a_1, a_2,…, a_n 互不相同。置换群的元素是置换，运算是置换的连接。置换的循环节

- 1 2 3 4 5
- =(134)(25) 这个置换有两个循环节
- 3 5 4 1 2

现在给定一些**特定**置换，形为 $a_i = (x+i) \bmod n + 1$ (x 为任意整数，i=1,2,…,n) 的{ a_n }

求其循环节的个数

如 n=2,x=1 则置换为 (2 1)=(12) 其循环节个数为 1

有 A,B,C 三根柱子和 n 个大小不一的圆盘

起初 n 个圆盘都插在 A 柱子上，而且遵循上小下大的顺序。

移动的规则：

每次只能移动一个盘子。

大盘子始终得放在小盘子下面。

求多少次移动可以将所有的盘子移到 C 柱上。

分析：T0=0

Tn=2Tn-1+1 (n>0)

解得

Tn=2n-1