算法案例难题-高中数学必修3第一章

1.辗转相除法

**例1** 求两个正数8251和6105的最大公约数.

（分析：8251与6105两数都比较大，而且没有明显的公约数，如能把它们都变小一点，根据已有的知识即可求出最大公约数）

解：8251＝6105×1＋2146

显然8251的最大公约数也必是2146的约数，同样6105与2146的公约数也必是8251的约数，所以8251与6105的最大公约数也是6105与2146的最大公约数.

6105＝2146×2＋1813，

2146＝1813×1＋333，

1813＝333×5＋148，

333＝148×2＋37，

148＝37×4＋0，

则37为8251与6105的最大公约数.

以上我们求最大公约数的方法就是辗转相除法.也叫欧几里德算法，它是由欧几里德在公元前300年左右首先提出的.利用辗转相除法求最大公约数的步骤如下：

第一步：用较大的数*m*除以较小的数*n*得到一个商*q*0和一个余数*r*0；

第二步：若*r*0＝0，则*n*为*m*，*n*的最大公约数；若*r*0≠0，则用除数*n*除以余数*r*0得到一个商*q*1和一个余数*r*1；

第三步：若*r*1＝0，则*r*1为*m*，*n*的最大公约数；若*r*1≠0，则用除数*r*0除以余数*r*1得到一个商*q*2和一个余数*r*2；

……

依次计算直至*rn*＝0，此时所得到的*rn*－1即为所求的最大公约数.

练习：利用辗转相除法求两数4081与20723的最大公约数.（答案：53）

2.更相减损术

我国早期也有解决求最大公约数问题的算法，就是更相减损术.

更相减损术求最大公约数的步骤如下：可半者半之，不可半者，副置分母·子之数，以少减多，更相减损，求其等也，以等数约之.

翻译出来为：

第一步：任意给出两个正数；判断它们是否都是偶数.若是，用2约简；若不是，执行第二步.

第二步：以较大的数减去较小的数，接着把较小的数与所得的差比较，并以大数减小数.继续这个操作，直到所得的数相等为止，则这个数（等数）就是所求的最大公约数.

**例2** 用更相减损术求98与63的最大公约数.

解：由于63不是偶数，把98和63以大数减小数，并辗转相减，即：

98－63＝35，

63－35＝28，

35－28＝7，

28－7＝21，

21－7＝14，

14－7＝7，

所以，98与63的最大公约数是7.

练习：用更相减损术求两个正数84与72的最大公约数.（答案：12）

3.比较辗转相除法与更相减损术的区别

（1）都是求最大公约数的方法，计算上辗转相除法以除法为主，更相减损术以减法为主，计算次数上辗转相除法计算次数相对较少，特别当两个数字大小区别较大时计算次数的区别较明显.

（2）从结果体现形式来看，辗转相除法体现结果是以相除余数为0则得到，而更相减损术则以减数与差相等而得到.

秦九韶计算多项式的方法



这就是我国南宋时期数学家秦九韶在他的著作《数书九章》中提出的算法.这种算法就叫秦九韶算法.

思考：对于*f*（*x*）＝（…（（*anx*＋*an*－1）*x*＋ *an*－2）*x*＋…＋*a*1）*x*＋*a*0，由内向外逐层计算一次多项式的值，其算法步骤如何？

第一步，计算*v*1＝*anx*＋*an*－1.

第二步，计算*v*2＝*v*1*x*＋*an*－2.

第三步，计算*v*3＝*v*2*x*＋*an*－3.

……

第*n*步，计算*vn*＝*vn*－1*x*＋*a*0.

思考：在秦九韶算法中，记*v*0＝*an*，那么第*k*步的算式是什么？

*vk*＝*vk*－1*x*＋*an*－*k*（*k*＝1，2，…，*n*）.

例1已知一个5次多项式为

,用秦九韶算法求这个多项式当时的值.

解： 根据秦九韶算法，把多项式改写成如下形式：

*f*（*x*）＝（（（（4*x*＋2）*x*＋3.5）*x*－2.6）*x*＋1.7）*x*－0.8

按照从内到外的顺序，依次计算一次多项式当*x*＝5时的值：

*v*0＝4；

*v*1＝4×5＋2＝22；

*v*2＝22×5＋3.5＝113.5；

*v*3＝113.5×5－2.6＝564.9；

*v*4＝564.9×5＋1.7＝2826.2；

*v*5＝2826.2×5－0.8＝14130.2.

所以，当*x*＝5时，多项式的值等于14130.2.

思考：（1）例1计算时需要多少次乘法计算？多少次加法计算？

（2）在利用秦九韶算法计算*n*次多项式当时需要多少次乘法计算和多少次加法计算？

练习：利用秦九韶算法计算



当时的值，并统计需要多少次乘法计算和多少次加法计算？

进位制是一种记数方式，用有限的[数字](http://zh.wikipedia.org/wiki/%E6%95%B0%E5%AD%97" \o "数字)在不同的位置表示不同的数值.可使用数字符号的个数称为基数，基数为*n*，即可称*n*进位制，简称*n*进制.现在最常用的是[十进制](http://zh.wikipedia.org/wiki/%E5%8D%81%E8%BF%9B%E5%88%B6" \o "十进制)，通常使用10个[阿拉伯数字](http://zh.wikipedia.org/w/index.php?title=%E9%98%BF%E6%8B%89%E4%BC%AF%E6%95%B8%E5%AD%97&action=edit" \o "阿拉伯數字)0－9进行记数.

对于任何一个数，我们可以用不同的进位制来表示.比如：[十进数](http://zh.wikipedia.org/wiki/%E5%8D%81%E8%BF%9B%E5%88%B6" \o "十进制)57，可以用[二进制](http://zh.wikipedia.org/wiki/%E4%BA%8C%E8%BF%9B%E5%88%B6" \o "二进制)表示为111001，也可以用[八进制](http://zh.wikipedia.org/wiki/%E5%85%AB%E8%BF%9B%E5%88%B6" \o "八进制)表示为71、用[十六进制](http://zh.wikipedia.org/wiki/%E5%8D%81%E5%85%AD%E8%BF%9B%E5%88%B6" \o "十六进制)表示为39，它们所代表的数值都是一样的.

表示各种进位制数一般在数字右下脚加注来表示，如111001（2）表示二进制数，34（5）表示5进制数.

电子计算机一般都使用二进制，下面我们来进行二进制与十进制之间的转化.

**例1** 把二进制数110011（2）化为十进制数.

解：110011＝1×25＋1×24＋0×23＋0×22＋1×21＋1×20

＝32＋16＋2＋1

＝51.

**例2** 把89化为二进制数.

解：根据二进制数满二进一的原则，可以用2连续去除89或所得商，然后取余数.

具体的计算方法如下：

因为89＝2×44＋1，

44＝2×22＋0，

22＝2×11＋0，

11＝2×5＋1，

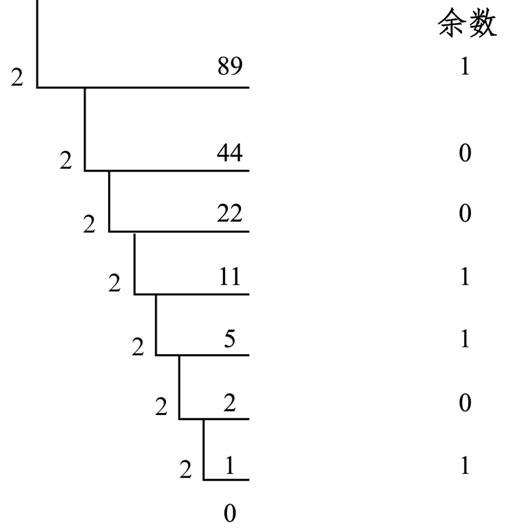
5＝2×2＋1，

所以，89＝2×（2×（2×（2×（2×2＋1）＋1）＋0）＋0）＋1

＝1×26＋0×25＋1×24＋1×23＋0×22＋0×21＋1×20

＝1011001（2）.

这种算法叫做除2取余法，还可以用下面的除法算式表示：



把上式中的各步所得的余数从下到上排列即可得到89＝1011001（2）

上述方法也可以推广为把十进制化为*k*进制数的算法，这种算法成为除*k*取余法.

当数字较小时，也可直接利用各进位制表示数的特点，都是以幂的形式来表示各位数字，比如2\*103表示千位数字是2，所以可以直接求出各位数字.即把89转换为二进制数时，直接观察得出89与64最接近故89＝64\*1＋25

同理：25＝16×1＋9，

9＝8×1＋1，

即89＝64×1＋16×1＋8×1＋1＝1×26＋1×24＋1×23＋1×20,

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 位数 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 数字 | 1 | 0 | 1 | 1 | 0 | 0 | 1 |

即89＝1011001（2）