第1章 整除 参考答案

计算证明

1. 计算下面整数对的最大公因子和最小公倍数.

（1）$(202, 282)$ 2，28482

（2）$(-666, 1410)$ 6，-156510

（3）$(30, 105, 360)$ 15，2520

（4）$(8n^2+28n+12, 12n^2+30n+12)$

$(8n^2+28n+12, 12n^2+30n+12) = (4(2n+1)(n+3), 6(2n+1)(n+2)) = (4,6)\cdot((2n+1)(n+3), (2n+1)(n+2))$ =2(2n+1)

$[8n^2+28n+12, 12n^2+30n+12] = 12(2n+1)(n+2)(n+3)$

2. 求下面整数的标准分解式.

（1）69 $69 = 3^1\times 23^1$

（2）200 $200 = 2^3\times 5^2$

（3）3288 $3288 = 2^3\times 3^1\times 137^1$

（4）22345680 $22345680 = 2^4\times 3^1\times 5^1\times 7^1\times 47^1\times 283^1$

3. 若 $a\in\mathbb{Z}^+\cup\{0\}$， a^4-3a^2+9 是质数还是合数?

解 $a^4-3a^2+9 = (a^2-3a+3)(a^2+3a+3) = [(a-1)(a-2)+1][(a+1)(a+2)+1]$

当 $a=0$， $a^4-3a^2+9=9\rightarrow$合数

当 $a=1$， $a^4-3a^2+9=7\rightarrow$质数

当 $a=2$， $a^4-3a^2+9=13\rightarrow$质数

当 $a>2$， $(a+1)(a+2)+1>(a-1)(a-2)+1>1\rightarrow$合数

综上所述，当 $a=1$ 或 $a=2$ 时为质数， 当 $a=0$或 $a>2$ 时为合数.

4. 若 $m-p\mid mn+pq$,求证 $m-p\mid mq+np$.

证明 易知 $m-p\mid(m-p)(n-q)$ ， 又 $m-p\mid mn+pq$ ， 而 $-(m-p)(n-q)+(mn+pq)=mq+np$ ， 则 $m-p\mid mq+np$ ， 证毕.

5. 设 $3 \mid a^2+b^2$，求证： $3 \mid a$ 且 $3 \mid b$.

证明 设 $a=3m+p,\ b=3n+q,\ (m,n\in\mathbb{Z},\ p,q\in\{-1,0,1\})$，

则 $a^2+b^2=(3m+p)^2+(3n+q)^2=3(3m^2+3n^2+2mp+2nq)+p^2+q^2$.

由 $3 \mid a^2+b^2$，得 $3 \mid p^2+q^2$，而 $p^2+q^2=\begin{cases}0, & p=q=0\\ 1, & p+q=\pm1\\ 2, & p=\pm1, q=\pm1\end{cases}$，故 $p=q=0$，$3 \mid a$ 且 $3 \mid b$，证毕.

6. 设 $a=qn-t$，若 $a \mid pm$，已知 $p-q=t$ 且 $(a,n+1)=1$，求证： $a \mid tm$.

证明 由 $a \mid pm$ 知 $a \mid pmn$，又 $a \mid am$，由整系数线性组合得 $a \mid pmn-am$，而 $a=qn-t$，则 $a \mid (p-q)m(n+1)$，又有 $p-q=t$ 且 $(a,n+1)=1$，得 $a \mid tm$，证毕.

7. 求证任意 n 个连续的正整数整数乘积都被 $n!$ 整除.（写出严谨的证明过程）

证明 数学归纳法.

记任意 n 个连续的正整数分别为 $a_0,a_1,\cdots,a_{n-1}$. 其中 $a_i=m+i\ (m\in\mathbb{N})$，乘积记为 T_m.

当 $m=1$ 时，$T_1=n!$，$n! \mid T_1$ 成立 .

假设当 $m=k\ (k\in\mathbb{N})$ 时，$n! \mid T_k$，即 $n! \mid k(k+1)\cdots(k+n-1)$ 成立 .

那么当 $m=k+1$ 时，有 $T_{k+1}=\prod_{i=0}^{n-1}a_i=(k+1)(k+2)\cdots(k+n)=k(k+1)\cdots(k+n-1)+n(k+1)\cdots(k+n-1)$.

只需证 $n! \mid n(k+1)\cdots(k+n-1)$，即$(n-1)! \mid (k+1)\cdots(k+n-1)$.

取 n 为 $n-1$，m 为 k，重复上述证明过程.

易知经过有限步后，$n=1$，而 $\forall m\in\mathbb{N},\ 1 \mid m$，假设成立，证毕.

8. 求证 $12 \mid n^4+2n^3+11n^2+10n,\ n\in\mathbb{Z}$.

证明 由$n^4+2n^3+11n^2+10n=(n-1)n(n+1)(n+2)+12n(n+1)$，只需证 $12 \mid (n-1)n(n+1)(n+2)$，易知 $[1,2,3,4] \mid (n-1)n(n+1)(n+2)$ （严谨证明可用**数学归纳法**），证毕.

9. 证明 n 的标准分解式中次数都是偶数当且仅当 n 是完全平方数 .

证明 充分性. 设 $n=m^2$，由算术基本定义有 $\begin{cases}n=p_1^{\alpha_1}\cdots p_s^{\alpha_s}, & p_i\text{是素数且}p_i<p_j(0<i<j\le s)\\ m=q_1^{\beta_1}\cdots q_t^{\beta_t}, & q_i\text{是素数且}q_i<q_j(0<i<j\le t)\end{cases}$，由算术分解的唯一性可得 $\begin{cases}s=t\\ p_i=q_i, & 0<i\le s\\ \alpha_i=2\beta_i, & 0<i\le s\end{cases}$. 必要性. 设 $n=p_1^{2\alpha_1}\cdots p_s^{2\alpha_s}$, p_i 是素数且 $p_i<p_j(0<i<j\le s)$，取 $m=p_1^{\alpha_1}\cdots p_s^{\alpha_s}$ 即可满足 $n=m^2$. 证毕.

10. 证明 $\sqrt{5}$ 是无理数. *并将其表示为简单连分数的形式.（*标注表示不作强制要求）

证明 反证法. 反设$\sqrt{5}=\dfrac{p}{q}$，其中 $p,q\in\mathbb{Z}^+$, $(p,q)=1$. 得 $p^2=5q^2$，则 $5 \mid p$.可设 $p=5m$，代入得$q^2=5m^2$，同理 $5 \mid q$. 则 $(p,q)=5\ne1$矛盾，故反设不成立，证毕.

对无理数$\alpha=\sqrt{5}$，有：

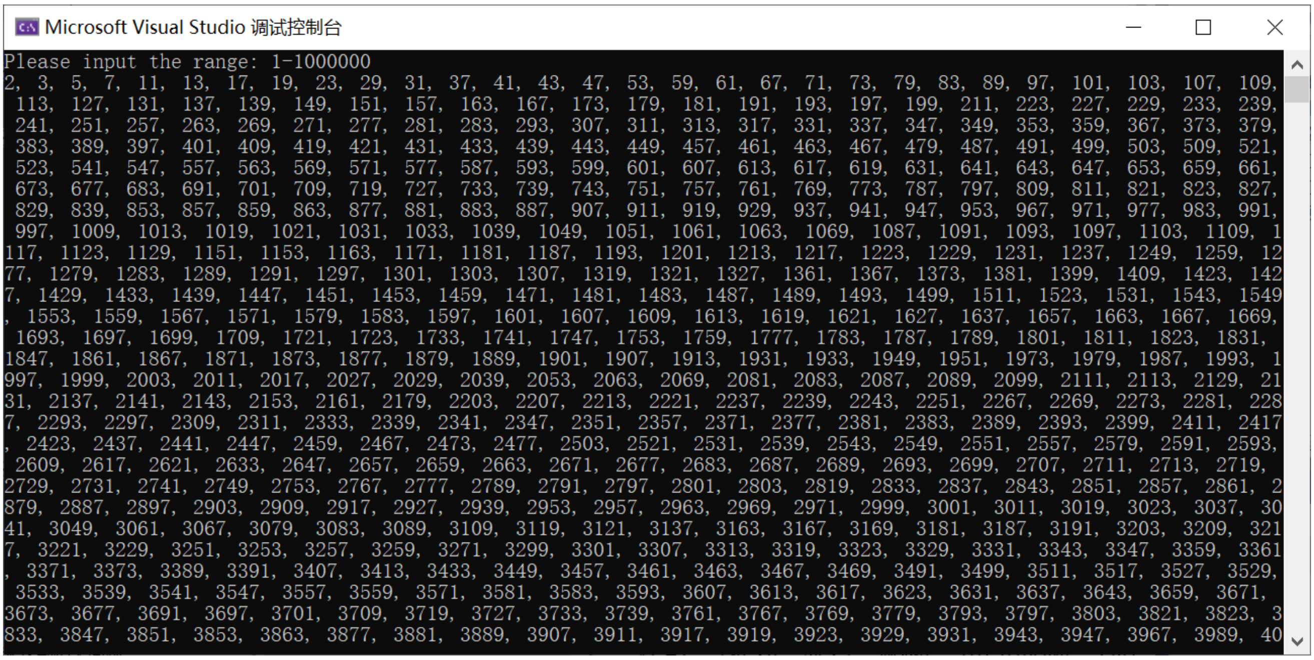
$$a_0=[\sqrt{5}]=2,\ \alpha_1=\frac{1}{\sqrt{5}-2}=\sqrt{5}+2$$

$$a_1=[\sqrt{5}+2]=4,\ \alpha_2=\frac{1}{\sqrt{5}+2-4}=\sqrt{5}+2=\alpha_1$$

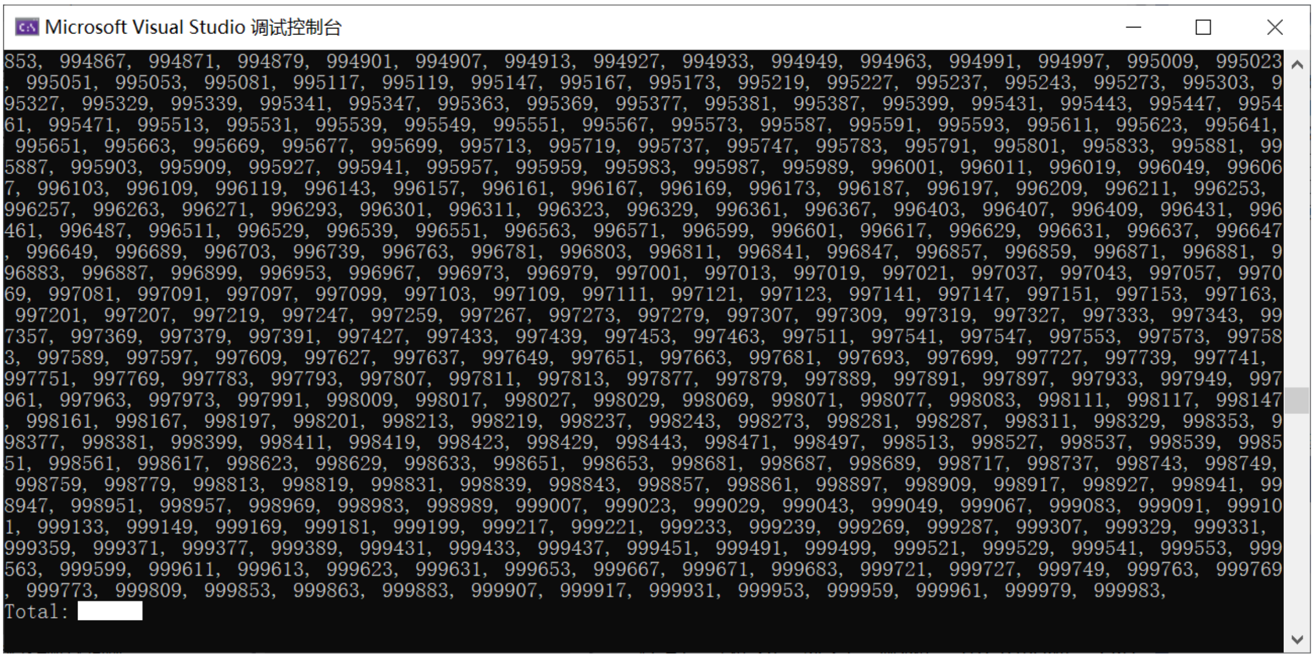
故 $\sqrt{5}=[2;\overline{4}]$.

编程练习（基于C/C++）

1. 编写程序使用Eratosthenes筛法打印1 000 000内所有素数及个数，效果如图所示．（*思考：a.对比筛法与普通算法的性能差异；b.递归调用该算法求更大范围素数进行优化；c.求更大的素数（如2^{512}数量级）该方法是否适用？会引入哪些新的问题？）


Microsoft Visual Studio 调试控制台

```
Please input the range: 1-1000000
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109,
 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239,
 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379,
 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467, 479, 487, 491, 499, 503, 509, 521,
 523, 541, 547, 557, 563, 569, 571, 577, 587, 593, 599, 601, 607, 613, 617, 619, 631, 641, 643, 647, 653, 659, 661,
 673, 677, 683, 691, 701, 709, 719, 727, 733, 739, 743, 751, 757, 761, 769, 773, 787, 797, 809, 811, 821, 823, 827,
 829, 839, 853, 857, 859, 863, 877, 881, 883, 887, 907, 911, 919, 929, 937, 941, 947, 953, 967, 971, 977, 983, 991,
 997, 1009, 1013, 1019, 1021, 1031, 1033, 1039, 1049, 1051, 1061, 1063, 1069, 1087, 1091, 1093, 1097, 1103, 1109, 1
117, 1123, 1129, 1151, 1153, 1163, 1171, 1181, 1187, 1193, 1201, 1213, 1217, 1223, 1229, 1231, 1237, 1249, 1259, 12
77, 1279, 1283, 1289, 1291, 1297, 1301, 1303, 1307, 1319, 1321, 1327, 1361, 1367, 1373, 1381, 1399, 1409, 1423, 142
7, 1429, 1433, 1439, 1447, 1451, 1453, 1459, 1471, 1481, 1483, 1487, 1489, 1493, 1499, 1511, 1523, 1531, 1543, 1549
, 1553, 1559, 1567, 1571, 1579, 1583, 1597, 1601, 1607, 1609, 1613, 1619, 1621, 1627, 1637, 1657, 1663, 1667, 1669,
 1693, 1697, 1699, 1709, 1721, 1723, 1733, 1741, 1747, 1753, 1759, 1777, 1783, 1787, 1789, 1801, 1811, 1823, 1831,
1847, 1861, 1867, 1871, 1873, 1877, 1879, 1889, 1901, 1907, 1913, 1931, 1933, 1949, 1951, 1973, 1979, 1987, 1993, 1
997, 1999, 2003, 2011, 2017, 2027, 2029, 2039, 2053, 2063, 2069, 2081, 2083, 2087, 2089, 2099, 2111, 2113, 2129, 21
31, 2137, 2141, 2143, 2153, 2161, 2179, 2203, 2207, 2213, 2221, 2237, 2239, 2243, 2251, 2267, 2269, 2273, 2281, 228
7, 2293, 2297, 2309, 2311, 2333, 2339, 2341, 2347, 2351, 2357, 2371, 2377, 2381, 2383, 2389, 2393, 2399, 2411, 2417
, 2423, 2437, 2441, 2447, 2459, 2467, 2473, 2477, 2503, 2521, 2531, 2539, 2543, 2549, 2551, 2557, 2579, 2591, 2593,
 2609, 2617, 2621, 2633, 2647, 2657, 2659, 2663, 2671, 2677, 2683, 2687, 2689, 2693, 2699, 2707, 2711, 2713, 2719,
2729, 2731, 2741, 2749, 2753, 2767, 2777, 2789, 2791, 2797, 2801, 2803, 2819, 2833, 2837, 2843, 2851, 2857, 2861, 2
879, 2887, 2897, 2903, 2909, 2917, 2927, 2939, 2953, 2957, 2963, 2969, 2971, 2999, 3001, 3011, 3019, 3023, 3037, 30
41, 3049, 3061, 3067, 3079, 3083, 3089, 3109, 3119, 3121, 3137, 3163, 3167, 3169, 3181, 3187, 3191, 3203, 3209, 321
7, 3221, 3229, 3251, 3253, 3257, 3259, 3271, 3299, 3301, 3307, 3313, 3319, 3323, 3329, 3331, 3343, 3347, 3359, 3361
, 3371, 3373, 3389, 3391, 3407, 3413, 3433, 3449, 3457, 3461, 3463, 3467, 3469, 3491, 3499, 3511, 3517, 3527, 3529,
 3533, 3539, 3541, 3547, 3557, 3559, 3571, 3581, 3583, 3593, 3607, 3613, 3617, 3623, 3631, 3637, 3643, 3659, 3671,
3673, 3677, 3691, 3697, 3701, 3709, 3719, 3727, 3733, 3739, 3761, 3767, 3769, 3779, 3793, 3797, 3803, 3821, 3823, 3
833, 3847, 3851, 3853, 3863, 3877, 3881, 3889, 3907, 3911, 3917, 3919, 3923, 3929, 3931, 3943, 3947, 3967, 3989, 40
```


Microsoft Visual Studio 调试控制台

```
853, 994867, 994871, 994879, 994901, 994907, 994913, 994927, 994933, 994949, 994963, 994991, 994997, 995009, 995023
, 995051, 995053, 995081, 995117, 995119, 995147, 995167, 995173, 995219, 995227, 995237, 995243, 995273, 995303, 9
95327, 995329, 995339, 995341, 995347, 995363, 995369, 995377, 995381, 995387, 995399, 995431, 995443, 995447, 9954
61, 995471, 995513, 995531, 995539, 995549, 995551, 995567, 995573, 995587, 995591, 995593, 995611, 995623, 995641,
 995651, 995663, 995669, 995677, 995699, 995713, 995719, 995737, 995747, 995783, 995791, 995801, 995833, 995881, 99
5887, 995903, 995909, 995927, 995941, 995957, 995959, 995983, 995987, 995989, 996001, 996011, 996019, 996049, 99606
7, 996103, 996109, 996119, 996143, 996157, 996161, 996167, 996169, 996173, 996187, 996197, 996209, 996211, 996253,
996257, 996263, 996271, 996293, 996301, 996311, 996323, 996329, 996361, 996367, 996403, 996407, 996409, 996431, 996
461, 996487, 996511, 996529, 996539, 996551, 996563, 996571, 996599, 996601, 996617, 996629, 996631, 996637, 996647
, 996649, 996689, 996703, 996739, 996763, 996781, 996803, 996811, 996841, 996847, 996857, 996859, 996871, 996881, 9
96883, 996887, 996899, 996953, 996967, 996973, 996979, 997001, 997013, 997019, 997021, 997037, 997043, 997057, 9970
69, 997081, 997091, 997097, 997099, 997103, 997109, 997111, 997121, 997123, 997141, 997147, 997151, 997153, 997163,
 997201, 997207, 997219, 997247, 997259, 997267, 997273, 997279, 997307, 997309, 997319, 997327, 997333, 997343, 99
7357, 997369, 997379, 997391, 997427, 997433, 997439, 997453, 997463, 997511, 997541, 997547, 997553, 997573, 99758
3, 997589, 997597, 997609, 997627, 997637, 997649, 997651, 997663, 997681, 997693, 997699, 997727, 997739, 997741,
997751, 997769, 997783, 997793, 997807, 997811, 997813, 997877, 997879, 997889, 997891, 997897, 997933, 997949, 997
961, 997963, 997973, 997991, 998009, 998017, 998027, 998029, 998069, 998071, 998077, 998083, 998111, 998117, 998147
, 998161, 998167, 998197, 998201, 998213, 998219, 998237, 998243, 998273, 998281, 998287, 998311, 998329, 998353, 9
98377, 998381, 998399, 998411, 998419, 998423, 998429, 998443, 998471, 998497, 998513, 998527, 998537, 998539, 9985
51, 998561, 998617, 998623, 998629, 998633, 998651, 998653, 998681, 998687, 998689, 998717, 998737, 998743, 998749,
 998759, 998779, 998813, 998819, 998831, 998839, 998843, 998857, 998861, 998897, 998909, 998917, 998927, 998941, 99
8947, 998951, 998957, 998969, 998983, 998989, 999007, 999023, 999029, 999043, 999049, 999067, 999083, 999091, 99910
1, 999133, 999149, 999169, 999181, 999199, 999217, 999221, 999233, 999239, 999269, 999287, 999307, 999329, 999331,
999359, 999371, 999377, 999389, 999431, 999433, 999437, 999451, 999491, 999499, 999521, 999529, 999541, 999553, 999
563, 999599, 999611, 999613, 999623, 999631, 999653, 999667, 999671, 999683, 999721, 999727, 999749, 999763, 999769
, 999773, 999809, 999853, 999863, 999883, 999907, 999917, 999931, 999953, 999959, 999961, 999979, 999983,
Total:
```

Total: 78498

```cpp
#include<iostream>
#include<math.h>
using namespace std;

bool is_prime(int n)
{
    if (n < 2)return false;
    bool flag = true;
    for (int i = 2; i < n; i++)
    {
        if (n % i == 0)
        {
            flag = false;
            break;
        }
    }
    return flag;
}

//based on Eratosthenes once
int find_prime(int range)
{
    int count = 0;
    bool* num = new bool[range + 1];
    for (int i = 1; i < range + 1; i++)
        num[i] = true;
    for (int i = 2; i < sqrt(range); i++)
    {
        if(is_prime(i))
        {
            for (int temp = i + i; temp <= range; temp += i)
                num[temp] = false;
        }
    }
    for (int i = 2; i < range + 1; i++)
    {
        if (num[i])
        {
            count++;
            cout << i << ", ";
        }
    }
    cout << endl;
    return count;
}

int main()
{
    int n;
    cout << "Please input the range: 1-";
    cin >> n;
    int count = find_prime(n);
    cout << "Total: " << count << endl;
    system("pause");
}
```

2. 编写程序计算最大公因数和最小公倍数，效果如图所示.

Microsoft Visual Studio 调试控制台

```
a=9876
b=6789
gcd(a,b)=3
lcm(a,b)=22349388
```

```cpp
#include<iostream>
using namespace std;

int Euclid(int a, int b)
{

    if (a < 0)a = -a;
    if (b < 0)b = -b;
    int r = a;
    while (r != 0)
    {
        r = a % b;
        a = b;
        b = r;
    }
    return a;
}

int main()
{
    int a, b;
    cout << "a=";
    cin >> a;
    cout << "b=";
    cin >> b;
    int gcd = Euclid(a, b);
    int lcm = a * b / gcd;
    cout << "gcd(a,b)=" << gcd << endl;
    cout << "lcm(a,b)=" << lcm << endl;
}
```

3. 编写程序实现算术基本定理，效果如下所示.

Microsoft Visual Studio 调试控制台

```
Please input n(n>0): 888
888=2^3*3^1*37^1
```

```cpp
#include<iostream>
#include<map>
using namespace std;
void num_decompose(int n)
{
    cout << n << "=";
    map<int, int>nums;
    while (n!=1)
    {
        int i = 2;
```

```cpp
        while (i <= n)
        {
            if (n % i == 0)
            {
                auto iter = nums.find(i);
                iter == nums.end() ? nums[i] = 1 : iter->second++;
                n /= i;
                break;
            }
            i++;
        }
    }

    for (auto iter = nums.begin(); iter != nums.end(); )
    {
        cout << iter->first << "^" << iter->second;
        if (++iter == nums.end())
            cout << endl;
        else
            cout << "*";
    }
}

int main()
{
    int n;
    cout << "Please input n(n>0): ";
    cin >> n;
    if (n == 1)
        cout << "1=1" << endl;
    else
        num_decompose(n);
    return 0;
}
```