

而 $4(n+1)! + n(n+1)(n+4) \equiv 2n(n+1) \equiv 4 \pmod{n+2} \Rightarrow n=2$ 矛盾, 假设不成立, $n+2$ 必为素数.

综上, 证毕.

编程练习 (基于C/C++)

1. 编程实现平方-乘算法, 效果如图所示.

Microsoft Visual Studio 调试控制台

```
Calculate a^n(mod m)...  
Please input:  
a=2021  
n=20212023  
m=2023  
2021^20212023(mod 2023)=671
```

```
1  #include<iostream>  
2  using namespace std;  
3  
4  int pow_mod(int a, int n, int m)  
5  {  
6      int rst = 1;  
7      while (n > 0)  
8      {  
9          if (n & 1)  
10         {  
11             rst *= a;  
12             rst %= m;  
13         }  
14         a *= a;  
15         a %= m;  
16         n >>= 1;  
17     }  
18     return rst;  
19 }  
20  
21 int main()  
22 {  
23     cout << "Calculate a^n(mod m)..." << endl;  
24     cout << "Please input:" << endl;  
25     int a, n, m;  
26     cout << "  a="; cin >> a;  
27     cout << "  n="; cin >> n;  
28     cout << "  m="; cin >> m;  
29     cout << a << "^" << n << "(mod " << m << ")=" << pow_mod(a, n, m) << endl;  
30     return 0;  
31 }
```

2. 编程实现扩展的欧几里得算法求逆元, 效果如图所示.

```
a=12345
b=65432
gcd(a, b)=1
lcm(a, b)=807758040
a-1=63561(mod 65432)
b-1=353(mod 12345)
```

```
1  #include<iostream>
2  using namespace std;
3
4  void swap(int& a, int& b)
5  {
6      a = a ^ b;
7      b = a ^ b;
8      a = a ^ b;
9  }
10
11 int extend_Euclid(int a, int b, int&inv_a, int&inv_b)
12 {
13     if (a < b) return extend_Euclid(b, a, inv_b, inv_a);
14     int a0 = a, b0 = b, q = 1;
15     int s0 = 1, s1 = 0, t0 = 0, t1 = 1;
16     while (a % b != 0)
17     {
18         q = a / b;
19         a = a % b;
20         swap(a, b);
21         s0 -= q * s1;
22         swap(s0, s1);
23         t0 -= q * t1;
24         swap(t0, t1);
25     }
26     inv_a = s1 > 0 ? s1 : s1 + b0;
27     inv_b = t1 > 0 ? t1 : t1 + a0;
28     return b;
29 }
30
31 int main()
32 {
33     int a, b, inv_a, inv_b;
34     cout << "a=";
35     cin >> a;
36     cout << "b=";
37     cin >> b;
38     int gcd = extend_Euclid(a, b, inv_a, inv_b);
39     int lcm = a * b / gcd;
40     cout << "gcd(a,b)=" << gcd << endl;
41     cout << "lcm(a,b)=" << lcm << endl;
42     cout << "a-1=" << inv_a << "(mod " << b << ")" << endl;
43     cout << "b-1=" << inv_b << "(mod " << a << ")" << endl;
44 }
```