# 第2次编程练习报告

姓名：齐明杰 学号：2113997 班级：信安二班

##### 编程练习1——平方乘算法

* **源码部分：**
* #include <bits/stdc++.h>
* **using** **namespace** std;
* **int** cal(**int** a,**int** n,**int** m){
* **if**(!n) **return** 1;
* **int** x = cal(a, n>>1, m);
* x = (x \* x) % m;
* **if**(n&1) **return** (x \* a) % m;
* **else** **return** x;
* }
* **int** main(){
* printf("Calculate a^n(mod m)...\nPlease input:\n");
* **int** a,n,m;
* printf("  a="); scanf("%d",&a);
* printf("  n="); scanf("%d",&n);
* printf("  m="); scanf("%d",&m);
* **int** ans = cal(a,n,m);
* printf("%d^%d(mod %d)=%d",a,n,m,ans);
* **return** 0;
* }
* **说明部分：**

在代码中，函数cal采用递归方式实现平方乘算法，输入参数包括底数a、指数n和模数m。当指数n为0时，返回1；否则，利用递归计算n/2的结果x，然后根据指数n的奇偶性进行平方和乘法运算，从而得到n的结果。

下面是该算法的详细步骤：

1.将指数n表示为二进制形式，例如n=(1011)2；

2.从高位到低位依次处理n的每一位;

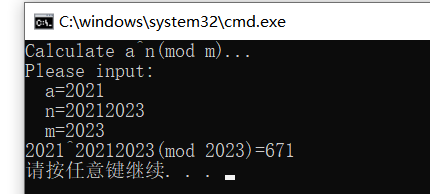
3.每次都将x平方并对m取模，即x=(x\*x)%m；

4.如果当前位为1，再将x乘以a并对m取模，即x=(x\*a)%m；

5.返回最终的x即可

因为n可以表示为二进制形式，所以平方乘算法的时间复杂度为O(log n)，比直接进行n次乘法的时间复杂度O(n)要快得多

* **运行示例：**



##### 编程练习2——拓展欧几里得算法

* **源码部分：**
* #include <bits/stdc++.h>
* **using** **namespace** std;
* **int** gcd(**int** a, **int** b){
* **if**(b==0) **return** a;
* **else** **return** gcd(b,a%b);
* }
* **int** lcm(**int** a, **int** b){
* **return** a\*b/gcd(a,b);
* }
* **int** inverse(**int** a,**int** m){
* **int** r, q, s1 = 1, s2 = 0, s3, t1 = 0, t2 = 1, t3 = 1;
* **while**(1){
* r = m%a; q = m/a;
* **if**(!r) **break**;
* m = a; a = r;
* s3 = s1 - q\*s2;
* t3 = t1 - q\*t2;
* s1 = s2; s2 = s3;
* t1 = t2; t2 = t3;
* }
* **return** t3;
* }
* **int** main(){
* **int** a,b;
* printf("a="); scanf("%d", &a);
* printf("b="); scanf("%d", &b);
* **int** g = gcd(a, b), l = lcm(a, b);
* printf("gcd(a,b)=%d\n", g);
* printf("lcm(a,b)=%d\n", l);
* **int** an = inverse(a%b,b), bn = inverse(b%a,a);
* an = an < 0 ? an + b : an;
* bn = bn < 0 ? bn + a : bn;
* printf("a^(-1)=%d(mod %d)\n", an, b);
* printf("b^(-1)=%d(mod %d)", bn, a);
* **return** 0;
* }
* **说明部分：**

通过不断迭代si,ti最终求得tn即可，因此可设s1,s2,s3,t1,t2,t3进行循环迭代，同时设r,q保存当前余数和除数,用m和a保存上一轮被除数和除数。

扩展欧几里得算法的具体实现步骤：

1.声明s1=1，s2=0，s3，t1=0，t2=1，t3，r，q；

2.循环执行以下步骤，直到余数为0：

(1)计算商q=m/a，余数r=m%a；

(2) 将被除数和除数转移到下一轮的值，即m=a,a=r

(3) 按公式计算s3、t3的值：s3=s1-q\*s2，t3=t1-q\*t2，然后将s1,s2,t1,t2转移到下一轮的值

3.如果计算结果an<0，则an+=m，返回a。

通过以上步骤可以求出a在模m下的乘法逆元an，如果an为负数，则将其加上m得到正数。

* **运行示例：**

