定理1.1 除法完整证明

> 定理1.1如下

对任意给定的整数 a 和 b，其中 b > 0，存在唯一的整数对 q（商）和 r（余数）使得，

$$a = qb + r$$

且 0 ≤ r < b。

构建集合

$$S = a - bk : k \in Z且a - bk \geq 0$$

> 由良序原则： 自然数的非空子集必然存在一个最小元素。

可得集合S中必有一最小元素r存在。

因

$$r = a - bk且k \in Z$$

可得存在唯一

$$q = k$$

当且仅当存在唯一r时成立。

由集合元素的不可重复性（互异性）可得r唯一

即

$$q = (a - r)/b$$

唯一

代码相关作业

```
void swap(int &a,int &b){
    int temp=a;
    a=b;
    b=temp;
}

int gcd(int a,int b)
{
    while(b){
        if(b>a) swap(a,b);
        else a=a%b;
    }
    return a;
```

```
}
//迭代gcd
```

```
long long multiply(int a,int b){
    long long ans=0;
    int n=0;//移位次数
    while(b!=1){
        if((b%2)==1){
            b=b>>1;
            ans+=a<<n;
        }
        else {
            b=b>>1;
            }
        n++;
    }
    ans+=a<<n;
    return ans;
}
//迭代版本简单乘法
```

```
void swap(int &a,int &b){
    int temp=a;
    a=b;
    b=temp;
}

int* egcd(int a,int b){
    int* fac=new int[2];
    int r0=1,r1=0,s0=0,s1=1;
    float p;
    if(a<b) swap(a,b);
    while(b){
        p=1.0*a/b;
        a=b;
        b=a%b;
        r0=r1;
        r1=r0-r1*p;
        s0=s1;
        s1=s0-s1*p;
    }
    fac[0]=r0,fac[1]=s1;
    return fac;
}
//egcd迭代版本
```

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

void swap(int &a,int &b){
    int temp=a;
    a=b;
    b=temp;
}

int gcd(int a,int b)
{
    while(b){
        if(b>a) swap(a,b);
        else a=a%b;
    }
    return a;
}
//迭代gcd

int main(){
    int n;
    cin>>n;
    int i=2,k=0;
    while(i<n){
        if(gcd(i,n)==1) k++;
        i++;
    }
    cout<<k;
    return 0;
}

//实现输出大于等于1，小于n，且与n互素的正整数的个数
```

第二章6、8题

第6题

证明：

对于

$$g \equiv 1(mod \quad m)$$

有

$$g^a \equiv 1^a(mod \quad m)$$

成立

那么，

$$g^a \equiv g^b \equiv g^{gcd(a,b)} \equiv 1(mod \quad m)$$

恒成立

第8题

证明：

不妨构建集合**S** 使得

$$S = k : (a \quad mod \quad k) = (b \quad mod \quad k) = 0$$

由

$$d = gcd(a, b)$$

对于

$$\frac{a}{d} \quad \frac{b}{d}$$

假设其公约数为x且x不等于1，则必有

$$d \quad mod \quad x * d \; x \in S且x \neq 1$$

d可以整除x*d不成立