逆元的几个求法。

1.扩展欧几里得$exgcd$

```
void exgcd(int a,int b,int &x,int &y){
    if(!b) x=1,y=0;
    else{
        exgcd(b,a%b,x,y);
        int tmp=x;
        x=y;
        y=tmp-(a/b)*y;
    }
}
///////更简洁的写法
void exgcd(int a,int b,int &x,int &y){
    if(!b) x=1,y=0;
    else exgcd(b,a%b,y,x),y-=a/b*x;//y1赋给x,x1赋给y,y=x1-(a/b)*y1 -> y=y-a/b*x;
}
```

2.费马小定理。

$(\frac{a}{b} \bmod p) = a \times b^{p-2} \bmod p \quad (p为质数,b不是p的倍数)$

3.线性递推。

$inv[1] = 1 \pmod p$,令$p = k \times i + r$

即：$k = p/i, r = p \bmod i$ (1)

$k \times i + r = 0 \pmod p$

等式两边同乘$inv[i], inv[r]$

$k \times inv[r] + inv[i] = 0 \pmod p$

$inv[i] = -k \times inv[r] \pmod p$

将(1)代入得：$inv[i] = -\frac{p}{i} \times inv[(p \bmod i)] \bmod p$

因为$-\frac{p}{i} mod\, p = (p - \frac{p}{i}) mod\, p$ (最小正整数解)

综上$inv[i] = (p - \frac{p}{i}) \times inv[(p\, mod\, i)] mod\, p$

这样就可以进行递推了。

4.阶乘逆元。

$facinv[i+1] = \frac{1}{(i+1)!}$

$facinv[i+1] \times (i+1) = \frac{1}{i!} = facinv[i]$

因此可以从后往前递推。也是线性的。

也可以从前往后推，不过要先求出$inv[i]$

$facinv[i] = facinv[i-1] \times inv[i] \bmod p$