扩展欧几里得复习篇。

由于多校考了欧几里得，所以这里复习一波扩欧。

这里主要讲解利用扩欧求解二元一次方程的整数解问题。

0.求解：$ax+by=gcd(a,b)$。

根据欧几里得知识可知：$gcd(a,b)=gcd(b,a\%b)$。

所以有式子:

$$\begin{cases} ax+by=gcd(a,b) \\ bx_1+(a\%b)y_1=gcd(b,a\%b) \end{cases}$$

即:$ax+by=bx+(a\%b)y.$

令:$a\%b=(a-\lfloor\frac{a}{b}\rfloor\times b)$。

则有:

$ax+by=bx_1+(a-\lfloor\frac{a}{b}\rfloor\times b)y_1$

$ax+by=ay_1+b(x_1-\lfloor\frac{a}{b}\rfloor y_1)$

即$x=y_1,y=(x_1-\lfloor\frac{a}{b}\rfloor y_1)$

显然我们只要知道了x_1,y_1,a,b就能求出x,y。因此不断往下递归。

因为我们知道$gcd(a,b)$最终会递归到$b=0$的时候，这是$gcd=a$。

所以对应$ax_k+by_k=gcd=a,b=0\rightarrow x_k=1,y_k=0.$

求出最底层的x_k,y_k后，我们就可以不断回溯，求出x,y了。

我们知道特解就可以求出通解了.

令$g=gcd(a,b),c=\frac{b}{g},d=\frac{a}{g}$

有$act=bdt=\frac{abt}{g}$，$t\in N.$

则通解$\begin{cases} x=x_0+ct \\ y=y_0-dt \end{cases}$

因为$a(x_0+ct)+b(y_0-dt)=ax_0+by_0+(act-bdt)=ax_0+by_0=gcd(a,b)$

1.求解：$ax+by=z$。

对应的：若$ax+by=z$有整数解，则$gcd(a,b)|z$，即$gcd(a,b)$能整除z。

因此要求解$ax+by=z$。

我们可以先求出$ax_1+by_1=gcd(a,b)$的解x_1,y_1。

令$t=\dfrac{z}{gcd(a,b)}$。

则$(ax_1+by_1)\times t=z\rightarrow a(tx_1)+b(ty_1)=z$.

即$x=tx_1,y=ty_1$。

因此该方程对应的特解我们就求出来,同理特解我们也能求出来。

2.求解：$ax+by=gcd(a,b)$的最小非负整数解。

因为我们可以得到通解$\begin{cases}x=x_0+ct\\y=y_0-dt\end{cases}$

所以我们只需要取适当的t就可以使x为非负数。

具体地:

$if(x<0)\ x=(x_0\%c+c)\%c$，这里用取模就很容易实现。

然后我们判断一下y是否为非负整数即可:$y=\dfrac{(gcd-ax)}{b}$.

3.具体实现的代码(两种写法).

```
void exgcd(int a,int b,int &x,int &y){
    if(!b){
        x=1,y=0;return;
    }
    exgcd(b,a%b,x,y);
    int tmp=x;
    x=y;
    y=tmp-(a/b)*y;
}
/////////////////////////////////////
void exgcd(int a,int b,int &x,int &y){
    if(!b){
        x=1,y=0;
        return;
    }
    exgcd(b,a%b,y,x); //这里仔细体会一下,这里我们将x1赋给y,y1赋给x
    y-=(a/b)*x;
}
```