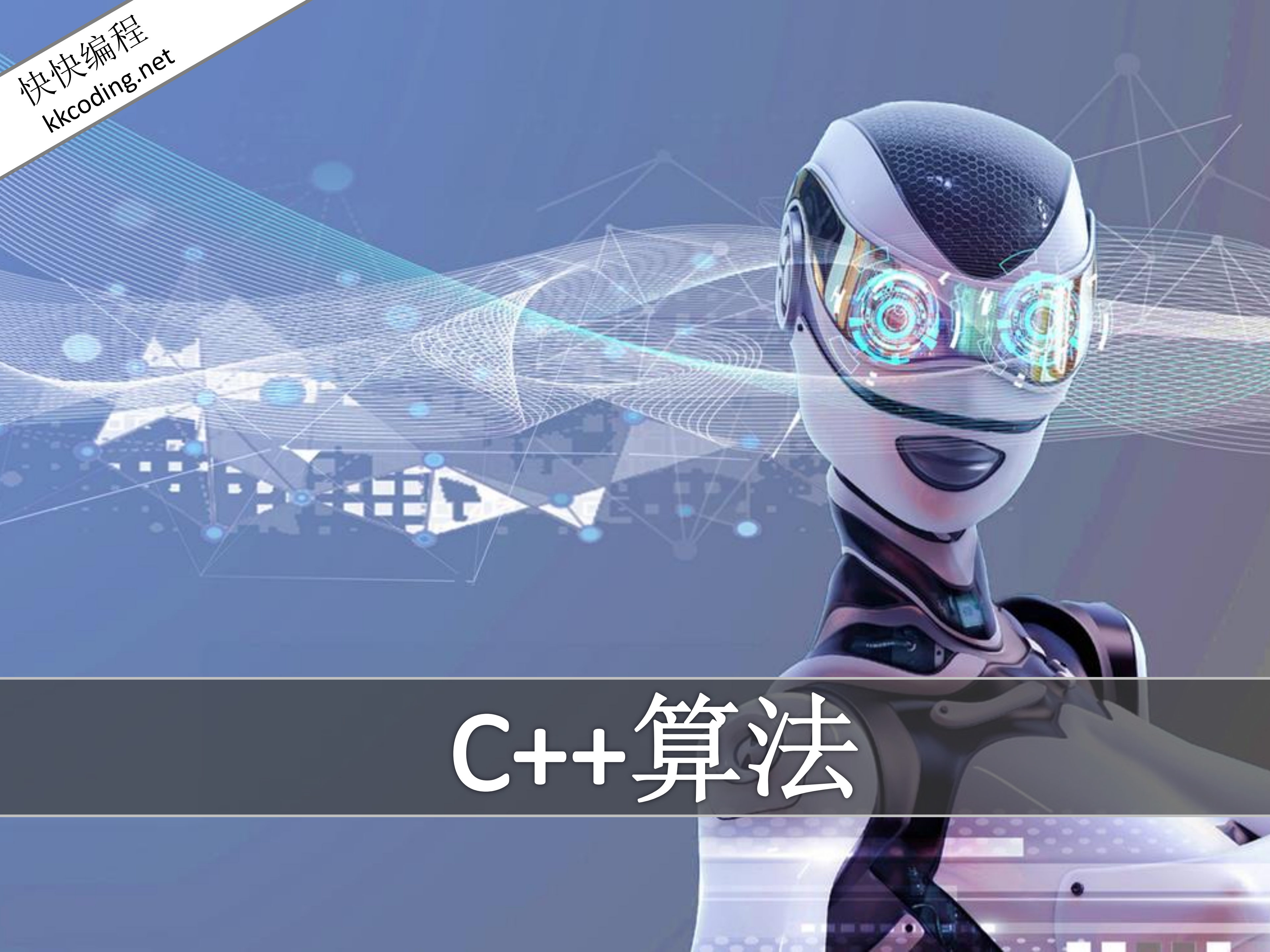
快快编程
kkcoding.net
C++算法

函数参数传递

引用传递
pass-by-reference

```cpp
#include<bits/stdc++.h>
using namespace std;
void passByValue(int x){
	x++;
}
void passByReference(int &y){
	y++;
}
int main(){
	int n=1;
	passByValue(n);
	cout<<n<<endl;
	passByReference(n);
	cout<<n<<endl;
	return 0;
}
```

拷贝
原变量n数值

为原变量n
重新命名为y

预测结果

1

2

模的世界

除法取余数

模5的世界	7的倍数

0	5	10	15	20	25	30	35	40	...
1	6	11	16	21	26	31	36	41	...
2	7	12	17	22	27	32	37	42	...
3	8	13	18	23	28	33	38	43	...
4	9	14	19	24	29	34	39	44	...

0*7,1*7,2*7,3*7,4*7,...

0*7%5,1*7%5,2*7%5,3*7%5,4*7%5,...

024130241302413...

周期性遍历	恰好访问所有 gcd(5,7)即1 的倍数 {0,1,2,3,4}

模a的世界
b的倍数有周期性遍历

若gcd(a,b)为1	互质
{0*b%a,1*b%a,2*b%a,...} 该集合恰好就是 {0,1,2,...,a-1}	

二元一次不定方程
ax+by=n 已知a,b,n求x,y

给定方程6x+8y=1
求出(x,y)的整数解

发现gcd(6,8)为2

6x+8y一定也是2的倍数

该方程无解

贝祖定理 Bezout	已知a,b 方程ax+by=n 有整数解 当且仅当 n是gcd(a,b)倍数
也叫 裴蜀定理	

	给定方程6x+8y=4 求出(x,y)的整数解
第一步	发现gcd(6,8)为2
第二步	给定方程3x+4y=2 求出(x,y)的整数解
第三步	给定方程3x+4y=1 求出(x,y)的整数解
	x=-1，y=1
第四步	答案乘2 x=-2，y=2
第五步	x=-2-4k，y=2+3k

求解算法

给定方程5x+7y=3
求出一组(x,y)的整数解

请提出算法步骤

先判断是否有解	求gcd

算法1	枚举x,范围0到6 判断3-5x是否为7的倍数

算法2	先求5x+7y=1的解
	辗转相除法
	扩展欧几里得算法 extended gcd

求解算法

辗转相除法

扩展欧几里得算法

extended gcd

ax+by=gcd(a,b)

观察辗转相除法过程

两式右侧相等

bx'+(a%b)y'=gcd(b,a%b)

$$bx' + \left(a - \left\lfloor \frac{a}{b} \right\rfloor b\right) y' = ax + by$$

根据a,b整理每项

$$ay' + b\left(x' - \left\lfloor \frac{a}{b} \right\rfloor y'\right) = ax + by$$

$$x = y', \qquad y = x' - \left\lfloor \frac{a}{b} \right\rfloor y'$$

由x',y'推导x,y

辗转相除法
扩展欧几里得算法 extended gcd
ax+by=gcd(a,b)

```
/*
辗转相除 (a,b) -> (b,a%b)
bx'+(a-a/b*b)y'=gcd(a,b)
根据a,b分类整理
ay'+b(x'-a/b*y')=gcd(a,b)
所以x=y',y=x'-a/b*y'
*/
```

$$x = y', \qquad y = x' - \left\lfloor \frac{a}{b} \right\rfloor y'$$

由x',y'
推导x,y

辗转相除法

扩展欧几里得算法
extended gcd

ax+by=gcd(a,b)

```
ll exgcd(ll a,ll b,ll&x,ll&y){
    if(b==0){
        x=1; y=0;
        return a;
    }
    ll xp,yp;
    ll g=exgcd(b,a%b,xp,yp);
    x=yp;
    y=
    return g;
}
```

$$x = y', \qquad y = x' - \left\lfloor \frac{a}{b} \right\rfloor y'$$

由x',y'
推导x,y

快快编程2327

```
min x+y
s.t. ax+by=n
     x,y>=0
```

数据规模：
对于10%数据，T=1,a=b
对于40%数据，T<=1000,1<=n,a,b<=10^4
对于100%数据，T<=100000,1<=n,a,b<=10^9

数据分类+多算法混合

```
int main(){
    freopen("steps.in","r",stdin);
    freopen("steps.out","w",stdout);
    ll T;
    cin>>T;
    while(T--){
        cin>>n>>a>>b;
        if(a<b) swap(a,b);
        if(a==b)
            solveEq();
        else if(n<=10000)
            solveBF();
        else
            solveMath();
    }
    return 0;
}
```

哪怕正解有错
其他部分可以得分

部分分

```
void solveEq(){
	if(n%a) cout<<-1<<" ";
	else cout<<n/a<<" ";
}
```

部分分

min x+y,s.t. ax+by=n, x,y>=0

假设a>=b,枚举x要x尽量大

```
void solveBF(){
	ll xbound=n/a;
	for(ll x=xbound;x>=0;--x){
		ll remain=n-a*x;
		if(remain%b)continue;
		ll y=remain/b;
		cout<<x+y<<" ";
		return;
	}
	cout<<-1<<" ";
}
```

时间复杂度?
$O(\sqrt{N})$

部分分

min x+y,s.t. ax+by=n, x,y>=0

假设a>=b,枚举x要x尽量大

确定有解

举例	31x+2y=100	枚举x=3,2,1,0
	a=31偏大	枚举量较小 不超过n/a+1

举例	3x+2y=100	枚举x=33,32
	a=3偏小	枚举量较小 不超过b=2<=a

最差情况:
n/a和a很接近

时间复杂度?

$O(\sqrt{N})$

完成1-17行
老师检查

```
1  /*姓名XXX
2  min x+y
3  s.t. ax+by=n
4       x,y>=0
5  若a>=b, 求 min y
6  求g=gcd(a,b,&x,&y)
7  n%g!=0无解
8  a/=g,b/=g,n/=g, 此时gcd(a,b)为1
9  ax+by=1,
10 a(x*n)+b(y*n)=n
11 x*=n, y*=n;
12 调整非负y尽量小
13 a(x-b) + b(y+a) =n,
14 y不断加a变成非负  if(y<0) y+=(-y)/a*a; if(y<0) y+=a;
15 y不断减a变成最小非负  y-=y/a*a;
16 x=(n-b*y)/a
17 */
```

快快编程445

Sylvester定理

已知正整数a和b互质,对于不同整数n,
方程ax+by=n解的存在性如下:

若n=ab-a-b,方程无非负整数解

若n=ab,方程无正整数解

若n>ab-a-b,方程有非负整数解

若n>ab,方程有正整数解

Sylvester定理

方程5x+7y=n解的存在性如下:

0	5	10	15	20	25	30	35	40	...
1	6	11	16	21	26	31	36	41	...
2	7	12	17	22	27	32	37	42	...
3	8	13	18	23	28	33	38	43	...
4	9	14	19	24	29	34	39	44	...

不能用ax+by表示的数里最大的是

$$b(a-1)-a=ab-a-b$$

数据分类+多算法混合

```
int main(){
    freopen("kai.in","r",stdin);
    freopen("kai.out","w",stdout);
    cin>>a>>b;
    if(a<b) swap(a,b);
    if(a<=1000)
        solveBF();
    else
        solveMath();
    return 0;
}
```

哪怕正解有错
其他部分可以得分

```
void solveBF(){
	ll cnt=1,ans;
	ok[0]=1;
	for(ll n=1;1;++n){
		if(OK(n,a,b)){
			ok[n]=1;
			cnt++;
		}else{
			cnt=0;
			ans=n;
		}
		if(cnt>=b){
			cout<<ans<<endl;
			return ;
		}
	}
}
```

枚举n,ok[n]判断ax+by=n
是否有非负整数解
设a>=b,
若连续b个ok[n]为1的话,
则后续全部有解

快快编程作业

2327

445

Sylvester定理(拓展题)

已知正整数a和b互质,共有几个正整数n, 使得方程ax+by=n无非负整数解?
答案=(a-1)(b-1)/2

证明思路1	{0,1,2,..,ab-a-b}共(a-1)(b-1)个数里 正好一半使方程有非负整数解 正好另一半使方程无非负整数解

证明思路2	总结"模世界"图表中规律

Sylvester定理(拓展题)

已知正整数a和b互质,共有几个正整数n, 使得方程ax+by=n无非负整数解?
答案=(a-1)(b-1)/2

0	5	10	15	20	25	30	35	40	...
1	6	11	16	21	26	31	36	41	...
2	7	12	17	22	27	32	37	42	...
3	8	13	18	23	28	33	38	43	...
4	9	14	19	24	29	34	39	44	...

证明思路2

$$\left\lfloor \frac{1\times7}{5} \right\rfloor + \left\lfloor \frac{2\times7}{5} \right\rfloor + \left\lfloor \frac{3\times7}{5} \right\rfloor + \left\lfloor \frac{4\times7}{5} \right\rfloor$$

Sylvester定理(拓展题)

$$A = \left\lfloor \frac{1\times7}{5} \right\rfloor + \left\lfloor \frac{2\times7}{5} \right\rfloor + \left\lfloor \frac{3\times7}{5} \right\rfloor + \left\lfloor \frac{4\times7}{5} \right\rfloor$$

难点：向下取整的处理

思考若没有向下取整，总和B是多少

$$B = \frac{1\times7}{5} + \frac{2\times7}{5} + \frac{3\times7}{5} + \frac{4\times7}{5} = \frac{7\times(5-1)}{2}$$

思考A和B每一项的差别的总和有多少

Sylvester定理(拓展题)

$$A = \left\lfloor\frac{1\times7}{5}\right\rfloor + \left\lfloor\frac{2\times7}{5}\right\rfloor + \left\lfloor\frac{3\times7}{5}\right\rfloor + \left\lfloor\frac{4\times7}{5}\right\rfloor$$

$$A = \left\lfloor\frac{4\times7}{5}\right\rfloor + \left\lfloor\frac{3\times7}{5}\right\rfloor + \left\lfloor\frac{2\times7}{5}\right\rfloor + \left\lfloor\frac{1\times7}{5}\right\rfloor$$

$$1 \times 7\%5 + 4 \times 7\%5 = 5$$

$$A = \left(\left\lfloor\frac{1\times7}{5}\right\rfloor + \left\lfloor\frac{4\times7}{5}\right\rfloor\right) + \left(\left\lfloor\frac{2\times7}{5}\right\rfloor + \left\lfloor\frac{3\times7}{5}\right\rfloor\right)$$

$$= \left(\frac{5\times7}{5} - 1\right) \times 2 = \frac{(7-1)(5-1)}{2}$$