BSGS

求解最小正整数x,满足$A^x \equiv B, 1 \leq A, B, p \leq 10^9$,$gcd(A,p)=1$
取$M=\sqrt{p}, x$表示为$aM-b$，转为求解
$A^{aM} \equiv B \times A^b (mod p)$
所以我们枚举b，把对应的$B \times A^b$扔到hash或者unordered_map里面，枚举左侧a，查询满足上述条件的b即可，$O(\sqrt{p})$
$0 \leq a, b \leq M$

```
//手写hash
const int HashMod=sqrt(1e9+10);
struct HashTable
{
    struct Line{int u,v,next;}e[1000000];
    int h[HashMod],cnt;
    void Add(int u,int v,int w){e[++cnt]=(Line)
{w,v,h[u]};h[u]=cnt;}
    void Clear(){memset(h,0,sizeof(h));cnt=0;}
    void Hash(int x,int k){
        int s=x%HashMod;
        Add(s,k,x);
    }
    int Query(int x){
        int s=x%HashMod;
        for(int i=h[s];i;i=e[i].next)
            if(e[i].u==x)return e[i].v;
        return -1;
    }
}Hash;
ll mypow(ll a,ll b,ll p){
    ll ans=1;
    while(b){
        if(b&1)ans=ans*a%p;
        a=a*a%p;
        b>>=1;
    }
    return ans;
}
int BSGS(int A,int B,int p)
{
    if(A%p==0){return -1;//无解}
    A%=p;B%=p;
```

```
    if(B==1){return 0;}
    int m=sqrt(p)+1;
    Hash.Clear();
    for(int i=0,t=B;i<m;++i,t=1ll*t*A%p)Hash.Hash(t,i);
    for(int i=1,tt=mypow(A,m,p),t=tt;i<=m+1;++i,t=1ll*t*tt%p){
        int k=Hash.Query(t);if(k==-1){continue; };
        return i*m-k;
    }
    return -1;//无解
}
```

```
//unorder_map版本
unordered_map<ll,ll>Map;
ll BSGS(ll A,ll B){
    Map.clear();
    ll m=sqrt(p)+1,tmp=0;
    if(A%p==0&&B==0)return 1;
    if(A%p==0&&B!=0)return -1;
    for(int i=0;i<=m;++i){
        if(!i){ tmp=B%p;Map[tmp]=i;continue;}
        tmp=(tmp*A)%p;
        Map[tmp]=i;
    }
    tmp=1;ll t=mypow(A,m);
    for(int i=1;i*i<=p;++i){
        tmp=(tmp*t)%p;
        if(Map[tmp]){
            ll ans=i*m-Map[tmp];
            return ans;
        }
    }
    return -1;//-1无解
}
```

Exbsgs

求解$A^x \equiv B(mod p)$ 的最小正整数,p 不为质数， 设$D = gcd(A,p)$,假如D不能整除B并且$B \neq 1$,则无解，所以

$$\frac{A^{x-1}A}{G} \equiv \frac{B}{G}(mod\frac{p}{G})$$,p'显然小于p不断换元递归，到p'变成质数的时候BSGS即可

```
int gcd(int a,int b){ return b?gcd(b,a%b):a;}
const int HashMod=123456;//质数
struct HashTable
{
    struct Line{int u,v,next;}e[1000000];
    int h[HashMod],cnt;
    void Add(int u,int v,int w){e[++cnt]=(Line)
{w,v,h[u]};h[u]=cnt;}
    void clear(){memset(h,0,sizeof(h));cnt=0;}
    void Hash(int x,int k){
        int s=x%HashMod;
        Add(s,k,x);
    }
    int Query(int x){
        int s=x%HashMod;
        for(int i=h[s];i;i=e[i].next)
            if(e[i].u==x)return e[i].v;
        return -1;
    }
}Hash;
int mypow(int a,int b,int mod){
    int ans=1;
    while(b){
        if(b&1)ans=1ll*ans*a%mod;
        a=1ll*a*a%mod;b>>=1;
    }
    return ans;
}
int exbsgs(int A,int B,int p){
    if(B==1){return 0; }
    int k=0,a=1;
    while(1){
        int d=gcd(A,p);
        if(d==1)break;
        if(B%d){ return -1;}//无解
        B/=d;p/=d;++k;
        a=1ll*a*A/d%p;
        if(B==a)return k;
    }
    Hash.clear();
    int m=sqrt(p)+1;
    for(int i=0,t=B;i<m;++i,t=1ll*t*A%p)Hash.Hash(t,i);
    for(int
i=1,tt=mypow(A,m,p),t=1ll*a*tt%p;i<=m;++i,t=1ll*t*tt%p){
        int x=Hash.Query(t);
```

```
        if(x==-1){continue; };
        return i*m-x+k;
    }
    return -1;//无解
}
```