# Big step

2417

/\*

(1) 求M = ceil( sqrt(C) ) ；

(2) for(i=0;i<M;i++) hash( i , A^i ) ；

(3) 求D = A^M%C；

(4) r = 1 ; for( i = 0 ; i < M ; i++ ){

ex\_gcd(r , C , x , y ) ;

res = x \* B % C ; jj = find( res);

}\*/

#include<cstdio>

using namespace std;

const int M = 1234567;

struct Node{

int num;

long long val;

}baby[M];

bool cmp(Node n1,Node n2){

return n1.val!=n2.val?n1.val<n2.val:n1.num<n2.num;

}

long long p,b,n,m;

long long powmod(long long A,long long B,long long mod){

long long tmp=1;

A%=mod;

while(B){

if(B&1)tmp=tmp\*A%mod;

A=A\*A%mod;

B=B>>1;

}

return tmp;

}

int div(long long val){

int low=0,high=m-1,mid;

while(low<=high){

mid=(low+high)>>1;

if(baby[mid].val==val)

return baby[mid].num;

if(baby[mid].val<val)

low=mid+1;

else

high=mid-1;

}

return -1;

}

int babystep(){

baby[0].num=0;baby[0].val=1;

for(int i=1;i<m;i++){

baby[i].num=i;

baby[i].val=(baby[i-1].val\*b)%p; //b^i

}

sort(baby,baby+m,cmp);

sort(baby,baby+m,cmp);

int cnt=1;

//去除余数相同但是标号大的

for(int i=1;i<m;i++) {

if(baby[i].val!=baby[cnt-1].val){

baby[cnt++]=baby[i];

//printf("%lld\n",baby[cnt-1].val);

}

}

long long bm=powmod(powmod(b,p-2,p),m,p);//先求逆元(费马小定理)，再求b^(-m)

long long ans=-1;

long long tmp=n;

for(int j=0;j<m;j++){ //查找(b^(-m))^j

int pos=div(tmp);

if(pos!=-1){

ans=j\*m+pos;

break;

}

tmp=(tmp\*bm)%p;

}

if(ans<0)

puts("no solution");

else

printf("%d\n",ans);

return 0;

}

int main(){

while(cin>>p>>b>>n){

m = (int)ceil(sqrt((double)(p))); //ceil向上取整

babystep();

}return 0;

}