2022.11.15

题号	期望分数	实际分数	误差
reverse	45	35	-10
silhouette	0	8	+8
seat	25	5	-20
ancient	40	10	-30

~~依旧拉的一批~~

T1

考场上打了一个寄搜，大样例跑了11s还是错误答案，但是小样例都过了，就错认为是数据范围打了导致寄搜出现意外，没想到其实是思路有亿点小问题(指没有考虑到会换到外面去)

正解

有几样东西是肯定的，s的答案是1，被限制的点的答案是-1；接下来就可以从s出发跑宽搜，观察一下翻转的性质，发现当前点要换到其他地方，只能换到和它距离奇数的点上，再考虑每次的左右边界，分别是$u-k+1$和$u+k-1$，但是直接这样会寄掉，要考虑到会不会换到外面去，所以还要再把区间给它锁死，理论上来说，需要用set优化才能到$O(nlogn)$的效率，但是数据比较水，手写队列也能过（`STL`过不了）（用`STL`卡常都过不了，还因为写了错误的快写de了很久）

```
#include<bits/stdc++.h>

using namespace std;
typedef long long ll;
ll re(){
    ll s=0,w=1;char c=getchar();
    while(!isdigit(c)){if(c=='-')w=-w;c=getchar();}
    while(isdigit(c)){s=(s<<1)+(s<<3)+(c^48);c=getchar();}
    return s*w;
}
inline int Max(int x,int y){return x>y?x:y;}
inline int Min(int x,int y){return x<y?x:y;}
const int D=1e5+114;
int n,s,k,m,dis[D];
bool dog[D];
int inf;
int q[D];
inline void bfs(){
    int l=1,r=0;
    q[++r]=s;
    while(l<=r){
        int u=q[l++];
        for(register int i=Max(u-k+1,k-u+1);i<=Min(u+k-1,(n*2)-k-u+1);i+=2){
            if(!dog[i]){
                dog[i]=1;
                dis[i]=dis[u]+1;
                q[++r]=i;
```

```
            }
        }
    }
}
int main()
{
    n=re();k=re();m=re();s=re();
    inf=0x3f3f3f3f;
    fill(dis+1,dis+1+n,inf);
    dis[s]=0;dog[s]=1;
    for(register int i=1;i<=m;i++){
        int a=re();
        dog[a]=1;dis[a]=-1;
    }
    bfs();
    for(register int i=1;i<=n;i++){
        if(!dog[i])dis[i]=-1;
        printf("%d ",dis[i]);
    }

    return 0;
}
```

T2

非常离谱的一道，考场上想了一个多小时也没想到暴力的思路，后来看题解也看不懂（**题解写的和雪一样）找好几个人问了之后才切掉（指用一晚上问题.jpg

分析

首先考虑更简单的去计算情况数，如果是一开始那样乱序的A, B，~~将对青少年产生不可估量的影响~~，会非常难去统计，因为不管$A，B$的顺序如何，对对方产生的影响都是不变的，所以我们可以将其排一手序，（我是从大到小，从小到大也可以，但是下面推式子得反着来）

在排序之后，我们用一个$a \times b$（a是A的个数，b同理）的矩形Z存下当前每个点，$Z_{i,j}$表示A中第i行，B中第j列交叉之后可以取到的最大值（因为我们已经排过序，所以Z中每一个值相同的区间，要么构成一个L型，要么构成一个矩形，先想如何处理一个矩形的情况

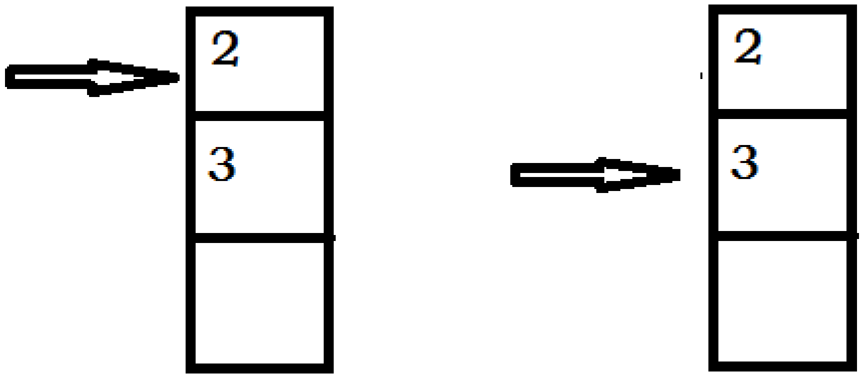
我们考虑让每一列都符合情况，设当前a行b列，设有至少i行不符合情况，当前区间最大值为w，此时我们的方案总数就是：

$$f(i) = C_a^i \times [w^i \times ((w+1)^{a-i} - w^{a-i})]^b$$

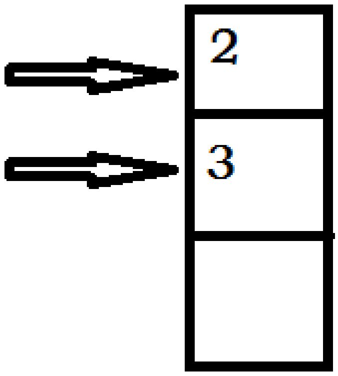
然后是L型，设上一次的边界为la和lb，当前扩展出的区间的长宽为a和b，则方案总数（画个图推一下就有了）为：

$$f(i) = C_a^i \times ([(w+1)^{la+a-i} - w^{la+a-i}]^b \times [(w+1)^{(a-i)^{lb}} \times w^{i^{lb+b}}])$$

一般的思考而言，很容易觉得$f(0)-f(1)$就是最后答案（剩下的为正确），但是！还要想到$f(1)$中可能出现重复的情况被减去，则答案会变少，举个栗子：



设当前有3行，钦定至少1行不满足，上界为4，如图是两种会被计算到最后$f(1)$中的情况，但是两种重合了，减去的时候也会多减，这时候我们就需要多加上一种：



而这种情况在$f(2)$中出现，且$f(2)$中也会有类似多加的情况，所以到$f(3)$时我们又需要减去，所以需要考虑一个容斥，那么要计算的值就是：

$$\sum_{i=0}^{a}(-1)^i \times f(i)$$

接下来计算就可以了

```
#include<bits/stdc++.h>

using namespace std;
typedef long long ll;
ll re(){
    ll s=0,w=1;char c=getchar();
    while(!isdigit(c)){if(c=='-')w=-w;c=getchar();}
    while(isdigit(c)){s=(s<<1)+(s<<3)+(c^48);c=getchar();}
    return s*w;
}
const ll mod=1e9+7;
const int D=5e6+114;
ll qp(ll a,ll b){
    ll res=1;
    while(b){
        if(b&1)res=(res*a)%mod;
        a=(a*a)%mod;b>>=1;
    }
    return (res+mod)%mod;
```

```
}
int n,cnt;
ll A[D],B[D],C[D<<1];
ll jc[D],inv[D];
void pre(){
    jc[0]=jc[1]=1;inv[0]=inv[1]=1;
    for(int i=2;i<=n;i++){
        jc[i]=(jc[i-1]*i)%mod;
    }
    inv[n]=qp(jc[n],mod-2);
    for(int i=n;i>=0;i--)inv[i-1]=inv[i]*i%mod;
    sort(A+1,A+n+1,greater<int>());
    sort(B+1,B+n+1,greater<int>());
    sort(C+1,C+cnt+1,greater<int>());
    cnt=unique(C+1,C+cnt+1)-C-1;
}
ll calc(int a,int b,int la,int lb,ll w){
    ll res=0,dog;
    for(int i=0;i<=a;i++){
        dog=1;
        dog=(dog*qp(qp(w+1ll,1ll*(la+a-i))-qp(w,1ll*(la+a-i)),1ll*b))%mod;
        dog=(dog*qp(w+1ll,1ll*lb*1ll*(a-i))%mod*qp(w,1ll*(lb+b)*i))%mod;
        dog=(dog*inv[a-i]%mod*inv[i])%mod;
        if(i&1)res=(res-dog+mod)%mod;
        else res=(res+dog+mod)%mod;
    }
    return res*jc[a]%mod;
}

int main()
{
    n=re();
    for(int i=1;i<=n;i++){A[i]=re();C[++cnt]=A[i];}
    for(int i=1;i<=n;i++){B[i]=re();C[++cnt]=B[i];}
    pre();
    ll ans=1;
    int la=0,lb=0,ra,rb;
    for(int i=1;i<=cnt;i++){
        ra=la;rb=lb;
        while(ra<n&&C[i]==A[ra+1])ra++;
        while(rb<n&&C[i]==B[rb+1])rb++;
        ans=(ans*calc(ra-la,rb-lb,la,lb,C[i]))%mod;
        if(ans==0)break;
        la=ra;lb=rb;
    }
    ans=(ans+mod)%mod;
    printf("%lld\n",ans);
    return 0;
}
```

T3

首先考虑了n=1和n=2的情况，然后看到有2^n-1的特殊情况，写了个特判去摸分（因为每次区间都是奇数长度，可以找出每次可能出现在哪些位置上），但是不知道那里写寄了（♠）

只有聪明人才能看懂的正解

桤? ? 栎? 悥旹癝兆(H岴癝9?劼 H泮H堷枳? 腇H翻H岴颒堌鑛d 隑H翻H岴餧堌鏾d ?H翻H岴豓堌鐳d

?H麔H岴騂墹?d ?H麔H岴驢墹?

祕E?E|-媥?E|%媥饗M媮 ??媥鲉M媮(??? ?? H葅0]肬H攵H汷0?EH媥H垗 ,E鶱黎 墹H?f? 袗H葅0]

肬H攵H汷0嵱塙D垗 H岴餟墹H?? 泄d H?/? 祕Ei? 媥拎墹玲)翂?聃E ?香?防i? 香?防拎A壚A拎A)繢壚?

香?防?羮彁H葅0]肬H攵嵱H塙L垗 ?]肬H攵H汷0嵱H塙L垗

蠬墂H媥H蚪H媥鳫?? H墹H亣? H咿t邇8/u贖H犥H墐 @?L9碵墡 ? H= ?u虾 苾 L嫪 H塇

? 傻8 ? 1译狤嫪 A嫯H= ? H岭H播 @?@埑 ?? M媡D夆H塇桠? L媥D夆H塇栌? 殂? H嫪 H?

紛 L?腜 ?€ H墂H尯H犥H墐 @?L9碵墡 剋 H= ?u虾 苾 L嫪 H塇 ? 傻8 ? 1译 ? 閁? I媡?

9卤? H婸?:區? M媥L媓H?鰤 ? L媟H媕I媞H?螃刌 H?赿 H塇钀? I僋 tD夆H塇M夎?4 ? H塇栉? M姓D

夆H塇枥? H呿勌? I夎D夆H塇栀3 楣? @ 莾0 棣? I媡?7吘? M媡I婭H?€zc? A?1? H壷L?0? ? L媡L

灵螃椓挄8製

A?甥 D夆H塇鑸3 L媥H?麃 ? I媡H? H瀧螃劌 ? H瀧L灵螃銄 H媥D夆H塇L媡?3 H媥?1咁? H媡儀呧?

H媡€8>吽? ? H塇璞? 檂?婂0 呉t荿葅H\A]A^A*肶?I尯嫚H鈚鑷 L**墻 肶L?I尯嫚H鈚柰 L**墻 牗H嫣**(H*

*茬**(嫚M媡H鈚?? @觵H壘(H媶 ujH?* L?_ ?H墂H尯劼 ?

 H塇?H I姓I墹? H塇?? H瀧H吋塲T?? 1篱? f? H塇D媖(D媜8媖@媎P椿 H婼€:I刪 H墣D媠(D塭8媠

@塲PH媍

(瑻 L媍

8H呹匃? 婼8;S<嶋? L婥0Lc蒋?K?葔S8??

H塇?? H瀧?? 禔<r勀 <p?? H秅H塊H塇桉E E1蕓壚篒 H塇柁? I壚镵? H塇栉? 尯呟? 塲T1篱膑

H塇?? H瀧榀? €yp叩? H屹H塁€yT卤 H秅1穑塊耍(;K,峢 Hc罘1纼?Hk?塊(HC H咿勚? ? 塒I壚槔?

婤湮?? 庲 1韮?凞 湮卄 H?笷 H夐鐵1 咿? €} n? 禘<at<w? 篿 H塇楢= H塇H軇鐻? H

婼I壞?<E豹 <p剳 <i哈? €zI呺? H塇瑗D I墹M夃? H塇钑? I夎I墹? H塇鑲? I姓I墹? H塇鐼? H瀧閰?

1鳑?區 H偣窾 H塇鐺= I姓I墹? H塇?? H瀧?? H婤H?香 H?媡夵CPH夐鏂0 咿刌 H婤媡湮剆 幀 湮

吋? H婤H? €xc刌 H塇柩C I壞H?\E H夐?0 咿刾 H?/H H夐?0 咿tH?H H夐梥/ 咿叓 H塇瑗 H軇H婥€8I

匄 I夐M夃? H塇鑋? I姓I墹? H塇鐿? H瀧镵? 咿兝? I姓E1珊4 H塇?? H瀧?? H塇?? 尯墂原? E1

篱~? H塇? I姓I墹? H塇桊? H瀧殛? H婤H?mD H?媡夵CPH亣?/ 咿勑 H婤媡湮劉 戀 湮劼 湮u臬?/D

H亣梏. 咿劆 €?n卝 禃<at<w匷 篿 H塇鐀; H塇H麔杳? H婼I壞?<E尯 <p劅 <i? €zI? H

塇?B I墹M夃? H塇?? I夬I墹? H塇柃? I姓I墹? H塇栍? I壚鑭? H秅H塊H塇?? H塇H瀧梚 H軇H婥

€8I? I夐I姓? H塇钼? H瀧閰? 禘 <mt<p卬? :E卐? H婥€8*劚 H**塇鑚**A ? I**壚 墹**H**塇鐼**? 锁**? 1**鰟**?**厰 H*

*偣窾 H**塇**?: I**姓**I**墹**? H**塇**?? I**壚閰**? H**塇鏈**? I**壚閰**? H**塇梵**? I**壚閰**? 婤湮凢**? 幛 1 湮**?? H**婤**H? €xc*

*刬 H**塇栊**@ I**壞**H?PB H**亣**?- 咿朥 H?#E H**亣**?- 咿**tH?E H**亣桉**, 咿銄 H**塇钏 H**麔**H**婥**€8I**勚 I**亣**M**夃**? H**塇**

鐕?*

*I**姓**I**墹**? H**塇鐲**? I**壚槊**? 咿**?? I**姓**E1**珊**4 H**塇**?? I**壚閮**? ?<mt<p**卄**? :G**匴**? H**婤**€8*刬 H塇桦? ? I壚I

墹H塇栎? ?? H塇杷? H塇H軇枥? H塇I壞璧? ?? H塇?? 閟? H塇杷 I夎I墹? H塇鏂? I墹楸? ? 峆

?v夵c<嚐? H塇枰? I壞緗? 窾 H塇鐼8 H軇楸? H秅H塊H塇瑗? H塇H瀧鍋 H麔H婥€8I勅 I亣I姓?

H塇?? I壚閿? H塇鏍? H瀧閳? H塇鐼? 锳? H塇柃> H軇閉? 窾

```
墹
堉曌 壚槭
媆
las_a=a,las_b=b;
}
printf("%lld\n",F_ans%MOD);
return 0;
}
```

@?L9頏埢 刓 H= ?u虾 苾 L嫋 H堉 ? 儍8 ? 1译狭婡H媢M吚tD夆H堉杌? H嫋 H= 劑 H

峆H播 l姓?{D夆苾 {H堉鑴? H嫋 H= 凛 H峆H播 ?}苾 }閅? M婡M吚tD夆H堉鐱? H儅 厛? H嫵

H倆? 嚶 H?霸 L?蛺 ? H姓H爈H亾H壏 D?L9鬞堝 ? H

? I9@劢 D夆H堉鑵? H嫇 H佲 ? H嵊H墐 ? 苾 f儅 匇 H?駞 H?餐 ?H壣H匏H爈H墐 @?H9頏埢 ??

H= ?u虾 苾 L嫋 H堉 ? 儍8 ? 1译括?瀰 H堉?? 轼? H堉箵 鏢? H婨D夆H堉L婡?? 钱 H

堉?? 锄? 儁吘? H婭€9>叡? ? H堉?? H婨H婡H?H婨

莽0 M婡D夆H堉鑷 ?? I媄?厓? l媢?)LD痖y? L婨D夆H堉? H?j H堉铇? ?? 綤 轭? 1? 橐? H婭龜?

M媄D夆H堉杪 M娄D夆H堉? 械? 壔X 莽0 梋? H婡? <0t^<1?? H?}i H堉?? 閹? L9鍁?*产 俏*

*啾 疸**H堉L**夋鏢***? 閧**? ? H**堉鍺**? L**婨槲**? H?!i H**堉枞**? ?? AUATUWVSH**洰**(A? 仒湮**wKL?k L**龘壼**H**壦*

*Ic**両栥 殽婨 H= 勧 H**峆**H**播 ? 夲苾 L**婨**H**堉**H**頔**([^]A\A]槐? H婨 H? L? ?fD H壣H匏H爈H墐

D?L9頓垡 勭 H= D?u魏 苾 L嫋 H堉 ? 儍8 ? 1译ゝ恷婡閉 € H婨 H?辭 H?郿 ?f? H壣

H匏H爈H墐 @?H9頏埢 割 H= ?u虾 苾 L嫋 H堉 ? 儍8 ? 1译? H婨 H?hi H?ji ?f? H壣H匏

H爈H墐 @?H9頏埢 ? H= ?u虾 苾 L嫋 H堉 ? 儍8 ? 1译? H婨 H?騢 H?鶆 ?f? H壣H匏H爈

H墐 @?H9頏埢 割 H= ?u虾 苾 L嫋 H堉 ? 儍8 ? 1译? H媄 H佲 劒 H嵊H墐 ? 苾

H= ? H峆H播 ?&苾 &@ H頔([^_]A\A]? H媄 H佲 劉 H嵊H墐 ? 苾 H?鯤 H?駁 ?fD H壣H匏H爈

H墐 @?H9頏埢 t嚨= ?u雍 苾 L嫋 H堉 ? 儍8 ? 1译?€ 幂� H婨 H= 凴 H峆H播 ?

*苾 ? f? H**婨 槼**? @ H**婨 ? @ H**婨 H?g H?g ?f? H**壣**H**匏**H**爈**H**墐 @?H9**頏埢 劒**? H= ?u*

*虾 苾 L**嫋 H**堉 ? 儍**8 ? 1**译**? H**婨 H? H?**╢ ?f? H**壣**H**匏**H**爈**H**墐 @?H9**頏埢 ?? H= ?u**虾*

*苾 L**嫋 H**堉 ? 儍**8 ? 1**译**? €? (t)H**婨 H= 劯 H**峆**H**播 ? 苾 L**婨夲**H**堉**H?**駒 H?**鴈 栩**? H**嫋 ?f?*

*H**壣**H**匏**H**爈**H**墐 @?H9**頏埢 刵**? H= ?u**虾 苾 L**嫋 H**堉 ? 儍**8 ? 1**译**? L**婨夲**H**堉鏀**? H**嫋 H=*

*劍 H**峆**H**播 ?)**苾)**轸**? @ ? 苾 L**嫋 H**堉 ? 儍**8 ? 1**篱 ? 茔 L**婨 ? 儍**8 ? 1**篱**? ? 茔 L*

*婨 ? 儍**8 ? 1**议**1? ? 苾 L**嫋 H**堉 ? 儍**8 ? 1**篱**/ ? 茔 L**婨 ? 儍**8 ? 1**议**6? ? 茔 L**婨*

*? 儍**8 ? 1**篱**? ? 茔 L**婨 ? 儍**8 ? 1**篱嘯 fD AWAVAUATUWVSH**洰**HM**吚**H**壩垚**L**壜**D**壣**u]**雅婥吚*

*uLL**婥**?A? u**峆鍙**?v9H**耍湮**)**夸 L**媫 H**壊 t>**湮*tS湮th變H鉳鏀? L墻 H?H呟t

Code

瑙 H岖螳墹?} 閼 H壝H岖癝墹鑷 ?H壝H岖緡墹?} H岖欵墹鐨 ?H壝H岖疕墹棙| H壼H墹棹 H壝H岖蠬墹? ?

H�THn 廿? H?瑁 H u 夂SH? 悵 ?H1 H? 瑶 H?? H H? H ? H H ?

H壈H屽道H墹柷| H屽緺墹? ?H壈H屽蝰墹瑛| H壺H墹瓒 忘葹h[]肬SH冹hH崿$€ H峪 H屽疕墹鑁| H崕疕屽營苪H? H墹鐣 H屽緁墹?| H崕緁屽癐苪H?涪 H墹? H崕欵屽瘹? A? H墹杩? H屽癛墹鑨 H屽

緁墹?| H屽欵墹鐳 H屽疕墹梏{ H?瞰 H?鄂 枢 H媮 H墹? H? H墹杵? H屽蝰墹鑺{ H崕蝰屽繧苪H?琛 H墹鑖 H屽道墹鐐{ H崕道屽蠹苪H?取 H墹鑚 H崕緺屽鳖? A? H墹?? H屽蠬墹瑧 H屽道墹鐵{ H屽緺

墹铓 H屽蝰墹?{ 闟 H壈H屽癛墹鑪 ?H壈H屽緁墹?{ H屽欵墹鑍 ?H壈H屽疕墹椼z H壺H墹桉 H壈H屽蠬墹? ?H壈H屽道墹枇z H屽緺墹? ?H壈H屽蝰墹瑜z H壺H墹瑭 忘葹h[]肬SH冹8H崿$€ H峪袎褎E?禘貓E 甀屽疕墹?z H崕疕屽甀苪H壆H婈需@ H屽疕墹鐳z ?H壈H屽疕墹?z H壺H墹? H媮蠬葹8[]肬H夊H冹 ?E? M?C? ?Y硫H,缐罥?1? 袗H葹]肬SH冹8H崿$€ H峪袎U鍖塃邢峪鏁屽 墹鑾y H屽 壚H?悫 H婈需? H屽 墹铚y 媮亓?踈鄿皬鑶E E? ?僻?H?皑 H婈需 僂?媮瑑?呏y迓}?囄 媮琀?? H?L? ?Hc蠬??? H?皅?g? H婈需? 僻?轟H?R? H婈需? 鎨H?C? H婈需? 儕H?4? H婈需? 隚H?%? H婈需v ?H?? H婈需d ?H?? H婈需R ?H? H婈需@ 愲僂?H?洄 H婈需) 媮?E箻潆1H壈H屽 墹鑠x H壺H墹鐪 H壈H婈需? H壺H墹鐶 H媮蠬葹8[]肬H夊H冹0H?D? 泻 H墹H?;? 蠬墕鳫媮鐠? 篳? H墹H?

蠬屽皅墹H?^? 需/? 壆苪柳凌睋?)睨犁?uO?? A壚篤UUUD壚麡D壚柳壯)翋??莔墹)罝媮旱彡D壚麡龙D壚柳)睨袓媮?袎E隨杳? 髫A壚篤UUUD壚麡D壚柳壯)翋??莔墹)彂E髫A壚旱彡D壚麡龙D壚柳)睨袓媮?袎E 鑠? 壆苪柳凌睋?)睨犁?uO鑺? A壚篤UUUD壚麡D壚柳壯)翋??莔

? ?? ?H?睖 ?? ?? 墹篻fff壢麡龙壢柳)睨辛???翋穛?麌牞tn?? 壆苪柳凌?兑)睨?独A? A? ? 墹H?:? 需蘼 壆苪柳凌?兑)睨?独A? A? ? 墹H?? 犁E?亇? ?? 荅 ?H?A? 鑺? 僂儅d~闒島嵞 H屽颒墹鐳f H屽颕壚 H?? H舥鏈? H屽颒墹鑲f H崀H泮H屽餟墹?f H屽餓壚H?皻 H夰?? H屽餟墹?f H亾H泮H屽醓墹柝e H屽醎壚 H?貟 H夰椏? H屽醓墹梡e H亾H泮H屽騂墹琚e H屽騃壚H?蠶 H夰瓔? H屽騂墹璋e H亾H泮H屽驢墹鑕e H屽驣壚H?厲 H夰鐨? H屽驢墹鑪e 荅 隩谷 H?コ需>? 墹篻fff壢麡漾壢柳)睨辛??翋溆屽癛c禛菱H蠬墹?c H墹楮? 僂 儅 d~ ?? 杷? 荅? 隣媮觥铝?醒鳴彂E

鳴铝?醒鳴蕒罥?? 祕E鼀?吚tH?鉤 鑵? ?H?鋥 鏡? 僂?儅龘~

T4 古代猪文

暴力思路很明显，盯着题目看一会儿就知道了直接计算$q^{\sum_{d|n} C_n^d}$

理论上来说能拿40pts，但因为正解是对幂次进行了降低，所以会寄（

~~不好评价~~

分析

很上流的数学题，反正不看题解我不会

因为q=999911659为质数，由欧拉定理的推论得：
$q^{\sum_{d|n} C_n^d} \equiv q^{\sum_{d|n} C_n^d \mod 999911658} (\mod 999911659)$

对999911658分解质因数，$999911658 = 2 \times 3 \times 4679 \times 35617$，发现它每一个质因子的指数都是1，那么我们可以枚举n的约数d，然后用Lucas定理求组合数C_n^d，然后分别计算出它们对四个质因子取模的结果，记录为a_1, a_2, a_3, a_4，为了方便计算，可以先处理出每个质因子p_i以内所有阶乘和其取模p_i的乘法逆元（必须每个质因子单独处理，因为取模的数不一样可能导致结果不同）

最后用中国剩余定理求解线性同余方程组

$$x \mod 2 = a_1$$

$$x \mod 3 = a_2$$

$$x \mod 4679 = a_3$$

$$x \mod 35617 = a_4$$

```
#include<bits/stdc++.h>
using namespace std;
typedef long long ll;
ll re(){
    ll s=0,w=1;char c=getchar();
    while(!isdigit(c)){if(c=='-')w=-w;c=getchar();}
    while(isdigit(c)){s=(s<<1)+(s<<3)+(c^48);c=getchar();}
    return s*w;
}
const int D=4e4;
const ll mod=999911659;
const ll ys[5]={2,3,4679,35617};
ll n,g;
ll ans,a[5];
ll jc[D],inv[D];
ll qp(ll a,ll b,ll p){
    ll res=1;
    while(b){
        if(b&1)res=(res*a)%p;
        a=(a*a)%p;b>>=1;
    }
    return res;
}
void pre(int lim){
    jc[0]=jc[1]=1;
    for(int i=2;i<=lim;i++){
        jc[i]=(jc[i-1]*i)%lim;
    }
}
ll C(ll x,ll y,ll p){
    if(x<y)return 0;
    return (jc[x]*qp(jc[y]*jc[x-y],p-2,p))%p;
}
ll Lucas(ll x,ll y,ll p){
    if(x==0)return 1;
    if(x<y)return 0;
    return Lucas(x/p,y/p,p)*C(x%p,y%p,p)%p;
}
void crt(){
    ll dog=mod-1;
    for(int i=0;i<4;i++){
        ans=(ans+a[i]*(dog/ys[i])%dog*qp((dog/ys[i]),ys[i]-2,ys[i]))%dog;
    }
}

int main()
{
    n=re();g=re();
    if(n==mod){puts("0");return 0;}
    for(int i=0;i<4;i++){pre(ys[i]);
```

```
        for(int j=1;j*j<=n;j++){
            if(n%j!=0)continue;
            a[i]=(a[i]+Lucas(n,j,ys[i]))%ys[i];
            if(j*j!=n)a[i]=(a[i]+Lucas(n,n/j,ys[i]))%ys[i];
        }
    }
    crt();
    ans=qp(g,ans,mod);
    printf("%lld\n",ans);
    return 0;
}
```

引理一：欧拉定理推论

若正整数a,n互质，则对于任意正整数b，有$a^b \equiv a^{b \mod fi(n)} \mod n$

引理二：Lucas定理

如果p是质数，则对于任意整数$1 \leq m \leq n$，有：

$$C_n^m \equiv C_{n \mod p}^{m \mod p} \times C_{n/p}^{m/p} \mod p$$

就是把n和m表示成p进制数，对p进制下的每一位分别计算组合数，最后再乘起来

引理三：中国剩余定理

设$m_1, m_2, m_3 \dots m_n$是两两互质的整数，$m = \Pi_{i-1}^n m_i, M = m/m_i$，$t_i$是线性同余方程$M_i t_i \equiv 1 \mod m_i$的一个解，对于任意n个整数$a_1, a_2, a_3 \dots a_n$，方程组：

$$x \equiv a_1(\mod m_1)$$

$$x \equiv a_2(\mod m_2)$$

$$...$$

$$x \equiv a_n(\mod m_n)$$

有整数解，解为$x = \sum_{i=1}^n a_i M_i t_i$

总·总结

感觉面对难题，实力还是非常贫弱，~~有巨大的提升空间~~，同时基础也不是很牢靠

而且像数学板块的知识，还不能熟练的组合运用，超级短板DA❤ZE