

2022 — 11 — 15 模拟赛总结

写在前言：即使知道自己要 *AFO* 了，也不要放弃，好吗，终局之战，就更该以最好的姿态背水一战。

题目	期望	实际	δ
<i>reverse</i>	80pts	91pts	+11pts
<i>silhouette</i>	5pts	5pts	0pts
<i>seat</i>	0pts	0pts	0pts
<i>ancient</i>	0pts	0pts	0pts

我好菜，怎么办，干不过，这样就完了吗？

T1

考场：

可以直接打暴力的说，直接找到合法的可扩展点暴力 *BFS* 求代价不就完了。

然而后来发现在特殊情况下会被卡到 $O(n^2)$ ，一时半会儿也没想出来怎么解决重复访问区间的问题（线段树优化建图或许可做？）。

改题：

起始可以用 *set* 分奇偶维护元素，对于已扩展过的元素直接删除就好了。迭代器有点小硬件，但还是搞明白了（大概），*set* 的小技巧 ++。

CODE

```
#include<bits/stdc++.h>
#define ll long long
using namespace std;
const ll inf=0x3f3f3f3f;
template<typename T>inline void read(T &x){
    x=0;
    char c=getchar();
    T ret=0;
    while(!isdigit(c))ret|=(c^'-'),c=getchar();
    while(isdigit(c))x=(x<<3)+(x<<1)+(c^48),c=getchar();
    if(ret)x=(~x)+1;
    return;
}
template<typename T>inline void print(T x){
    if(x<0)putchar('-'),x=(~x)+1;
    if(x>9)print(x/10);
    putchar((x-x/10*10)^48);
    return;
}
template<typename T>inline void wr1(T x){
    print(x);
```

```

    putchar(' ');
    return;
}
template<typename T>inline void wr2(T x){
    print(x);
    putchar('\n');
    return;
}
int n,k,m,s,dis[114514];
bool vis[114514];
set<int>tr[2];
priority_queue<pair<int,int>>q;
inline void dj(){
    q.push({0,s});
    while(!q.empty()){
        int x=q.top().second;
        q.pop();
        if(vis[x])continue;
        vis[x]=1;
        int now=(x&1)^(k&1);
        int L=max(1,x-(k-1)),R=min(x+(k-1),n);
        int l=L+(L+k-1)-x,r=R+(R-k+1)-x;// (对于1和n的情况,考虑长度为k的串左右端在1,
n的可扩展端点)
        for(auto i=tr[now].lower_bound(l);i!=tr[now].end()&&(*i)
<=r;tr[now].erase(i++)){
            if(dis[*i]>dis[x]+1){
                dis[*i]=dis[x]+1;
                q.push({-dis[*i],*i});
            }
        }
    }
}
int main(){
    freopen("reverse.in","r",stdin);
    freopen("reverse.out","w",stdout);
    read(n);
    read(k);
    read(m);
    read(s);
    memset(dis,0x3f,sizeof dis);
    for(int i=1;i<=m;++i){
        int x;
        read(x);
        dis[x]=-1;
    }
    for(int i=1;i<=n;++i){
        if(s==i||dis[i]==-1)continue;
        if(i&1){
            tr[0].insert(i);
        }
        else{
            tr[1].insert(i);
        }
    }
    dis[s]=0;

```

```

dj();
for(int i=1;i<=n;++i){
    if(dis[i]!=inf){
        wr1(dis[i]);
    }
    else{
        wr1(-1);
    }
}
return 0;
}

```

T2

考场：

我知道总方案怎么求（没什么用）。

想不到怎么去删除不合法答案，于是特判了 $k = 1$ 的点就跑了，（实际上浪费了很多时间，再次感受到自己的弱小）。

改题：

~~什么鬼东西，好恶心。~~

又是个不给人做的容斥题。

对于这道题，我们要求以下方程的解的个数： $\forall i \in [1, n], \max_{j=1}^n x_{i,j} = A_i, \max_{j=1}^n x_{j,i} = B_i$ 我们可以先将 A, B 排序，因为这样对结果没有影响，简单证明一下：先来考虑列，对于每一个 A_i ，无论哪一列在前，哪一列在后，最终所有列在第 i 行的最大值都需要是 A_i ；行同理。如果最大的 A_i 不等于最大的 B_i ，那么无解，否则一定有解。然后我们从大到小枚举 A, B 中每一个值 S ，对于排好序后每一个 S ，新扩展的区域要么是一个矩形，就是一个 L 型。

To be continued.....

CODE

```

#include<bits/stdc++.h>
#define ll long long
using namespace std;
const ll mod=1e9+7;
template<typename T>inline void read(T &x){
    x=0;
    char c=getchar();
    T ret=0;
    while(!isdigit(c))ret|=(c^'-'),c=getchar();
    while(isdigit(c))x=(x<<3)+(x<<1)+(c^48),c=getchar();
    if(ret)x=(~x)+1;
    return;
}
template<typename T>inline void print(T x){
    if(x<0)putchar('-'),x=(~x)+1;
    if(x>9)print(x/10);
    putchar((x-x/10*10)^48);
}

```

```

        return;
    }
    template<typename T>inline void wr1(T x){
        print(x);
        putchar(' ');
        return;
    }
    template<typename T>inline void wr2(T x){
        print(x);
        putchar('\n');
        return;
    }
    ll n,a[114514],b[114514],jc[114514],ny[114514],ans=1,mix[214514],cnt;
    inline bool cmp(ll a,ll b){
        return a>b;
    }
    inline ll ksm(ll x,ll y){
        ll t=1;
        while(y){
            if(y&1)t=t*x%mod;
            x=x*x%mod;
            y>>=1;
        }
        return t;
    }
    inline ll C(ll m,ll n){
        if(m>n)return 0;
        if(m==n||m==0)return 1;
        return jc[n]*ny[n-m]%mod*ny[m]%mod;
    }
    inline ll work1(ll a,ll b,ll la,ll lb,ll s){
        ll ans=0;
        for(int i=0;i<=a;++i){
            ll now=ksm(ksm(s,i)*((ksm(s+1,la+a-i)-ksm(s,la+a-i)+mod)%mod)%mod,b)%mod*ksm(ksm(s+1,a-i)*ksm(s,i)%mod,lb)%mod*C(i,a)%mod;
            if(i&1)ans=(ans-now+mod)%mod;
            else ans=(ans+now)%mod;
        }
        return ans;
    }
    int main(){
        freopen("silhouette.in","r",stdin);
        freopen("silhouette.out","w",stdout);
        read(n);
        jc[0]=ny[0]=1;
        for(int i=1;i<=n;++i){
            read(a[i]);
            mix[++cnt]=a[i];
            jc[i]=jc[i-1]*i%mod;
        }
        ny[n]=ksm(jc[n],mod-2);
        for(int i=n;i>=1;--i){
            ny[i-1]=ny[i]*i%mod;
        }
        sort(a+1,a+n+1);
    }

```

```

for(int i=1;i<=n;++i){
    read(b[i]);
    mix[++cnt]=b[i];
}
sort(b+1,b+n+1);
sort(mix+1,mix+cnt+1);
cnt=unique(mix+1,mix+cnt+1)-mix-1;
if(a[n]!=b[n]){
    wr2(0);
    return 0;
}
int ra=n+1,rb=n+1,lb=n,la=n;
for(int i=cnt;i>=1;--i){
    while(la>1&&a[la-1]==mix[i]){
        la--;
    }
    while(lb>1&&b[lb-1]==mix[i]){
        lb--;
    }
    ans=ans*work1(ra-la,rb-lb,n-ra+1,n-rb+1,mix[i])%mod;
    ra=la;
    rb=lb;
}
wr1(ans);
return 0;
}

```

T3

什么数学期望概率DP。~~不会不会，摆子。~~

其实是没时间改了。

T4

考场：

好像是书上原题？无所谓了，数学岂是我能驾驭的？

花了 20min 研究题面之后才开始写，大概的写了个 *lucas*（虽然是错的），可惜终究是错付了。

改题：

其实对于次方可以使用扩展欧拉定理消次。

因为 999911659 是质数， $\phi(999911659) = 999911658 = 2 \times 3 \times 4679 \times 35617$ ，~~记得最后一个质数写错了改了好久。~~

对于给定的 n ，我们找到其所有因数，分别在模上述 4 个质数意义下计算出 C_n^i 的值之和，用 CRT 求出最小非负整数解，最后用个 *ksm* 计算答案即可。

CODE

```

#include<bits/stdc++.h>
#define ll long long
using namespace std;
const ll mod=999911659;

```

```

const ll mod2=mod-1;
const ll prime[]={1,2,3,4679,35617};
template<typename T>inline void read(T &x){
    x=0;
    char c=getchar();
    T ret=0;
    while(!isdigit(c))ret|=(c^'-'),c=getchar();
    while(isdigit(c))x=(x<<3)+(x<<1)+(c^48),c=getchar();
    if(ret)x=(~x)+1;
    return;
}
template<typename T>inline void print(T x){
    if(x<0)putchar('-'),x=(~x)+1;
    if(x>9)print(x/10);
    putchar((x-x/10*10)^48);
    return;
}
template<typename T>inline void wr1(T x){
    print(x);
    putchar(' ');
    return;
}
template<typename T>inline void wr2(T x){
    print(x);
    putchar('\n');
    return;
}
ll n,g,jc[6][40000],ny[6][40000],sum[6];
ll C(ll m,ll n,ll id){
    if(m>n)return 0;
    if(m==n||m==0)return 1;
    return jc[id][n]*ny[id][n-m]%prime[id]*ny[id][m]%prime[id];
}
ll lucas(ll m,ll n,int id){
    if(m<=prime[id]&& n<=prime[id]){
        return C(m,n,id);
    }
    return C(m%prime[id],n%prime[id],id)*lucas(m/prime[id],n/prime[id],id)%prime[id];
}
ll exgcd(ll a,ll b,ll &x,ll &y){
    if(b==0){
        x=1;
        y=0;
        return a;
    }
    ll d=exgcd(b,a%b,x,y);
    ll z=x;
    x=y;
    y=z-y*(a/b);
    return d;
}
ll ksm(ll x,ll y,ll MOD){
    ll t=1;
    while(y){

```

```

        if(y&1)t=t*x%MOD;
        x=x*x%MOD;
        y>>=1;
    }
    return t;
}
int main(){
    freopen("ancient.in","r",stdin);
    freopen("ancient.out","w",stdout);
    read(n);
    read(g);
    if(g==mod){
        wr1(0);
        return 0;
    }
    else{
        for(int i=1;i<=4;++i){
            jc[i][0]=ny[i][0]=1;
            for(int j=1;j<=prime[i];++j){
                jc[i][j]=jc[i][j-1]*j%prime[i];
                ny[i][j]=ksm(jc[i][j],prime[i]-2,prime[i]);
            }
        }
        for(int i=1;i*i<=n;++i){
            if(n%i==0){
                if(i*i==n){
                    sum[1]=((sum[1]+lucas(i,n,1))%mod)%mod;
                    sum[2]=((sum[2]+lucas(i,n,2))%mod)%mod;
                    sum[3]=((sum[3]+lucas(i,n,3))%mod)%mod;
                    sum[4]=((sum[4]+lucas(i,n,4))%mod)%mod;
                    continue;
                }
                sum[1]=((sum[1]+lucas(i,n,1))%mod+lucas(n/i,n,1))%mod;
                sum[2]=((sum[2]+lucas(i,n,2))%mod+lucas(n/i,n,2))%mod;
                sum[3]=((sum[3]+lucas(i,n,3))%mod+lucas(n/i,n,3))%mod;
                sum[4]=((sum[4]+lucas(i,n,4))%mod+lucas(n/i,n,4))%mod;
            }
        }
        ll x=0;
        for(int i=1;i<=4;++i){
            ll a=mod2,b=prime[i],X,Y;
            ll d=exgcd(a/b,b,X,Y);
            x=(x+sum[i]*(a/b)%mod2*x%mod2+mod2)%mod2;
        }
        wr1(ksm(g,x,mod));
    }
    return 0;
}

```

