比赛链接

补题情况

题目	蒋贤蒙	王赵安	王智彪
A	0	0	0
B	0	0	0
C	2	0	0
D	0	0	0
F	0	0	0
G	0	0	0
I	0	0	2
J	0	0	2

题解

C. Cells

题意

给定一个二维平面，求满足如下条件的 n 元路径组个数：

1. 第 i 条路径 $(a_i,0)\to (0,i)$
2. 每次移动只能选择 $(x,y)\to (x-1,y),(x,y+1)$

数据保证 $a_{i-1}\lt a_i$

题解

显然有

$$ M= \begin{bmatrix} {a_1+1\choose 1}&{a_1+2\choose 2}&\cdots&{a_1+n\choose n}\\ {a_2+1\choose 1}&{a_2+2\choose 2}&\cdots&{a_2+n\choose n}\\ \vdots&\vdots&\ddots&\vdots\\ {a_n+1\choose 1}&{a_n+2\choose 2}&\cdots&{a_n+n\choose n}\\ \end{bmatrix} = \prod_{i=1}^n \frac 1{i!} \begin{bmatrix} \frac {(a_1+1)!}{a_1!}&\frac {(a_1+2)!}{a_1!}&\cdots&\frac {(a_1+n)!}{a_1!}\\ \frac {(a_2+1)!}{a_2!}&\frac {(a_2+2)!}{a_2!}&\cdots&\frac {(a_2+n)!}{a_2!}\\ \vdots&\vdots&\ddots&\vdots\\ \frac {(a_n+1)!}{a_n!}&\frac {(a_n+2)!}{a_n!}&\cdots&\frac {(a_n+n)!}{a_n!}\\ \end{bmatrix} $$

设 $x_i=a_i+1$则

$$ M= \prod_{i=1}^n \frac 1{i!} \begin{bmatrix} x_1&x_1(x_1+1)&\cdots&\prod_{i=0}^{n-1}(x_1+i)\\ x_2&x_2(x_2+1)&\cdots&\prod_{i=0}^{n-1}(x_2+i)\\ \vdots&\vdots&\ddots&\vdots\\ x_n&x_n(x_n+1)&\cdots&\prod_{i=0}^{n-1}(x_n+i)\\ \end{bmatrix} $$

从左到右用列消元，可以得到

$$ M= \prod_{i=1}^n \frac 1{i!} \begin{bmatrix} x_1&x_1^2&\cdots&x_1^n\\ x_2&x_2^2&\cdots&x_2^n\\ \vdots&\vdots&\ddots&\vdots\\ x_n&x_n^2&\cdots&x_n^n\\ \end{bmatrix} $$

每行都提出一个 x_i，就可以得到一个范德蒙行列式，于是有

$$ \det M=\prod_{i=1}^n \frac {a_i+1}{i!}\prod_{1\le i\lt j\le n}(a_j-a_i) $$

考虑 NTT 计算每个值在 $\prod_{1\le i\lt j\le n}(a_j-a_i)$ 出现的次数。

具体的，可以设 $f(x)=\sum_{i=1}^n x^{a_i},g(x)=\sum_{i=1}^n x^{-a_i}$，则每个值 k 出现次数就是 $[x^k]f(x)g(x)$。

注意还有 $i\lt j$ 的限制，根据对称性，考虑 $$然后做快速幂即可，时间复杂度 $O(V\log V)$。

```
const int mod=998244353,MAXN=1e6+5;
int quick_pow(int n,int k){
    int ans=1;
    while(k){
        if(k&1)ans=1LL*ans*n%mod;
        n=1LL*n*n%mod;
        k>>=1;
    }
    return ans;
}
namespace Poly{
    const int Mod=998244353,G=3;
    int rev[MAXN<<2],Wn[30][2];
    void init(){
        int m=Mod-1,lg2=0;
        while(m%2==0)m>>=1,lg2++;
        Wn[lg2][1]=quick_pow(G,m);
        Wn[lg2][0]=quick_pow(Wn[lg2][1],Mod-2);
        while(lg2){
            m<<=1,lg2--;
            Wn[lg2][0]=1LL*Wn[lg2+1][0]*Wn[lg2+1][0]%Mod;
            Wn[lg2][1]=1LL*Wn[lg2+1][1]*Wn[lg2+1][1]%Mod;
        }
    }
    int build(int k){
        int n,pos=0;
        while((1<<pos)<=k)pos++;
        n=1<<pos;
        _for(i,0,n)rev[i]=(rev[i>>1]>>1)|((i&1)<<(pos-1));
        return n;
    }
    void NTT(int *f,int n,bool type){
        _for(i,0,n)if(i<rev[i])
        swap(f[i],f[rev[i]]);
```

```
        int t1,t2;
        for(int i=1,lg2=0;i<n;i<<=1,lg2++){
            int w=Wn[lg2+1][type];
            for(int j=0;j<n;j+=(i<<1)){
                int cur=1;
                _for(k,j,j+i){
                    t1=f[k],t2=1LL*cur*f[k+i]%Mod;
                    f[k]=(t1+t2)%Mod,f[k+i]=(t1-t2)%Mod;
                    cur=1LL*cur*w%Mod;
                }
            }
        }
        if(!type){
            int div=quick_pow(n,Mod-2);
            _for(i,0,n)f[i]=(1LL*f[i]*div%Mod+Mod)%Mod;
        }
    }
    void mul(int *f,int _n,int *g,int _m){
        int n=build(_n+_m-2);
        _for(i,_n,n)f[i]=0;_for(i,_m,n)g[i]=0;
        NTT(f,n,1);NTT(g,n,1);
        _for(i,0,n)f[i]=1LL*f[i]*g[i]%Mod;
        NTT(f,n,0);
    }
}
int frac[MAXN],invf[MAXN],a[MAXN<<2],b[MAXN<<2];
int main(){
    frac[0]=1;
    _for(i,1,MAXN)
    frac[i]=1LL*frac[i-1]*i%mod;
    invf[MAXN-1]=quick_pow(frac[MAXN-1],mod-2);
    for(int i=MAXN-1;i;i--)
    invf[i-1]=1LL*invf[i]*i%mod;
    int n=read_int(),base=1e6,ans=1;
    _rep(i,1,n){
        int t=read_int();
        ans=1LL*ans*(t+1)%mod*invf[i]%mod;
        a[t]++;
        b[base-t]++;
    }
    Poly::init();
    Poly::mul(a,base+1,b,base+1);
    _rep(i,base+1,base*2)
    ans=1LL*ans*quick_pow(i-base,a[i])%mod;
    enter((ans+mod)%mod);
    return 0;
}
```

I. Incentive Model

题意

有两个人争夺 n 个物品，每一轮争夺一个物品，每次争夺都有成功概率。给定 x,y 代表初始双方的 $stake$ 分别为 ${\frac x y}$ 和 $1-{\frac x y}$ □每次争夺双方成功概率为自己的 $stake$ 比双方相加的 $stake$ □然后每一轮如果第一个人获胜，他之后的 $stake$ 要加上 w □问第一个人争夺成功轮数的期望，结果对 998244353 取模。

题解

我们设对于第一个人，第 i 轮获胜的概率为 X^{i} □第 i 轮的 $stake$ 期望为 S_{i} □

我们有 $S_{0}={\frac x y}$

然后我们有 $X_{i+1}={\frac {S_{i}} {1+w×i}}$ □ $S_{i+1}=S{i}+w×X_{i+1}$

所以有 $S_{i+1}=S_{i}+w×({\frac {S_{i}} {1+w×i}})=S_{i}×(1+{\frac w {1+w×i}})=S_{i}×{\frac {1+(i+1)×w} {1+i×w}}$

所以有 ${\frac {S_{i+1}} {1+(i+1)×w}}={\frac {S_{i}} {1+i×w}}=...={\frac {S_{0}} {1+0×i}}={\frac x y}$

所以有 $S_{n}={\frac x y}×(1+n×w)$

又因为获胜轮数可以表示为 ${\frac {S_{n}-{\frac x y}} {w}}$

所以化简得到答案为 ${\frac x y}×n$

```
#include <bits/stdc++.h>
#define int long long
using namespace std;
const int mod=998244353;
int qpow(int a,int b,int m=mod) {
    int r=1;
    while(b) {
        if (b&1) r=r*a%m;
        a=a*a%m,b>>=1;
    }
    return r;
}
int n,x,y,w;
#undef int

int main() {
    scanf("%lld %lld %lld %lld",&n,&w,&x,&y);
    printf("%lld\n",n*x%mod*qpow(y,mod-2)%mod);
    return 0;
```

```
}
```