

# 2020.06.01-2020.06.07 周报

## 团队训练

本周暂无

## \_wzx27

### 1.补题

补了一下之前比赛的题

#### [HDU6589 Sequence](#)

把数列  $a_i$  写成其生成函数的形式  $f(x)=\sum a_ix^i$  每个操作  $k$  相当于  $f(x)\cdot \sum x^{ik}$  由交换律知顺序不重要，所以可以统计每种操作的次数  $m_i$  最后有

$$f(x)\cdot (\sum x^i)^{m_1} \cdot (\sum x^{2i})^{m_2} \cdot (\sum x^{3i})^{m_3}$$

其中  $(\sum x^{ki})^m = \sum \binom{n+i-1}{i} x^{ik}$

另外模数刚好是998244353，做三次NTT即可

```
#include<bits/stdc++.h>
#define ll long long
#define pii_ pair<int,int>
#define mp_ make_pair
#define pb push_back
#define fi first
#define se second
#define rep(i,a,b) for(int i=(a);i<=(b);i++)
#define per(i,a,b) for(int i=(a);i>=(b);i--)
#define show1(a) cout<<#a<<" = "<<a<<endl
#define show2(a,b) cout<<#a<<" = "<<a<<" ; "<<#b<<" = "<<b<<endl
using namespace std;
const ll INF = 1LL<<60;
const int inf = 1<<30;
const int maxn = 2e6+5;
const ll M = 998244353;
inline void fastio() {ios::sync_with_stdio(false);cin.tie(0);cout.tie(0);}
ll qpow(ll a,ll b) {ll
s=1;while(b){if(b&1)s=(s*a)%M;a=(a*a)%M;b>>=1;}return s; }
int n,m,lim,l,r[maxn],cnt[maxn];
ll A[maxn],B[maxn],fac[maxn],inv[maxn];
int c[5];
void NTT(ll a[],int type)
```

```
{
    rep(i,1,lim-1) if(i<r[i]) swap(a[i],a[r[i]]);
    for(int mid=1;mid<lim;mid<=<=1){
        ll wn = qpow(3,(M-1)/mid/2);
        if(type== -1) wn = qpow(wn,M-2);
        for(int R=mid<<1,j=0;j<lim;j+=R){
            ll w = 1;
            for(int k=0;k<mid;k++,w=w*wn%M){
                ll x=a[j+k],y=w*a[j+mid+k]%M;
                a[j+k] = (x+y)%M;
                a[j+mid+k] = (x-y+M)%M;
            }
        }
    }
    if(type== -1){
        ll INV = qpow(lim,M-2);
        rep(i,0,lim) a[i] = a[i]*INV%M;
    }
}
ll Comb(ll a,ll b) {return a<b?0:fac[a]*inv[b]%M*inv[a-b]%M;}
void init()
{
    fac[0] = 1;
    rep(i,1,2000000) fac[i]=fac[i-1]*i%M;
    inv[2000000] = qpow(fac[2000000],M-2);
    per(i,2000000-1,0) inv[i]=inv[i+1]*(i+1)%M;
}
int main()
{
    fastio();
    int _;
    init();
    for(cin>>_;;_--){
        cin>>n>>m;
        memset(A,0,sizeof(A));
        memset(B,0,sizeof(B));
        rep(i,0,n-1) cin>>A[i];
        lim=1,l=0;
        while(lim<=2*n) lim<=<=1,l++;
        rep(i,0,lim-1) r[i] = (r[i>>1]>>1) | ((i&1)<<(l-1));
        cnt[1]=cnt[2]=cnt[3]=0;
        while(m--) {int x;cin>>x;cnt[x]++;}
        rep(i,1,3)if(cnt[i]){
            memset(B,0,sizeof(B));
            for(int j=0;j*i<n;j++) B[i*j] = Comb(cnt[i]+j-1,j);
            NTT(A,1);NTT(B,1);
            rep(i,0,lim) A[i] = A[i]*B[i]%M;
            NTT(A,-1);
            rep(i,n+1,lim) A[i]=0;
        }
    }
}
```

```

    }
    ll ans = 0;
    rep(i,0,n-1) ans ^= ((i+1)*A[i]);
    cout<<ans<<endl;
}
return 0;
}

```

## 2. 整数分拆

生成函数的基本应用，求  $x^1+x^2+x^3+\dots+x^k=n$  的非负整数解得个数。

### 2.1 有序分拆

把每个数写成生成函数  $(1+x+x^2+\dots)$  的形式

那么分拆数就等于  $(1+x+x^2+\dots)^k$  的  $n$  次项前面的系数  $\binom{n+k-1}{k-1}$

这个证明可以通过  $(1+x+x^2+\dots)^k = \frac{1}{(1-x)^k}$  的逐项求导证明

### 2.2 无序分拆

无序的分拆容易想到朴素  $O(n^2)$  的  $\text{dp}$

记  $P(n)$  为  $n$  的无序拆分数如果用生成函数来做就是

$$\sum_{i=1}^{\infty} P(i)x^i = (1+x+x^2+\dots)(1+x^2+x^4+\dots)\dots = \prod_{i=0}^{\infty} \frac{1}{1-x^{i+1}}$$

结合五边形数定理  $\prod_{i=0}^{\infty} (1-x^{i+1}) = 1 + \sum_{i=1}^{\infty} (-1)^i (x^{\frac{i(3i-1)}{2}} + x^{\frac{i(3i+1)}{2}})$  得到：

$$(1+P(1)x+P(2)x^2+\dots)(1-x-x^2+x^5+\dots) = 1$$

比较两边的系数得到  $P(n)-P(n-1)-P(n-2)+P(n-5)=0$  即  $P(n)-P(n-1)-P(n-2)+P(n-5)+P(n-7)-\dots = 0$

这样就得到了  $P(n) = \sum_{i=1}^{\infty} (-1)^i (P(n-\frac{i(3i-1)}{2}) + P(n-\frac{i(3i+1)}{2}))$

### HDU4651 模板题

```

#include<bits/stdc++.h>
#define ll long long
#define pii_ pair<int,int>
#define mp_ make_pair
#define pb push_back
#define fi first
#define se second
#define rep(i,a,b) for(int i=(a);i<=(b);i++)

```

```
#define per(i,a,b) for(int i=(a);i>=(b);i--)
#define show1(a) cout<<#a<<" = "<<a<<endl
#define show2(a,b) cout<<#a<<" = "<<a<<" ; "<<#b<<" = "<<b<<endl
using namespace std;
const ll INF = 1LL<<60;
const int inf = 1<<30;
const int maxn = 2e5+5;
const ll M = 1e9+7;
inline void fastio() {ios::sync_with_stdio(false);cin.tie(0);cout.tie(0);}

ll f1[maxn],f2[maxn],P[maxn];
void init()
{
    int n = 1e5;
    for(int i=1;;i++){
        f1[i] = (3*i*i-i)>>1;
        if(f1[i]>n) break;
        f2[i] = (3*i*i+i)>>1;
        if(f2[i]>n) break;
    }
    P[0] = 1;
    rep(i,1,n){
        for(int j=1;;j++){
            if(f1[j]<=i) P[i] += (j&1)?P[i-f1[j]] : -P[i-f1[j]];
            else break;
            if(f2[j]<=i) P[i] += (j&1)?P[i-f2[j]] : -P[i-f2[j]];
            else break;
        }
        P[i] = (P[i]%M+M)%M;
    }
}
int main()
{
    fastio();
    init();
    int _,n;
    for(cin>>_;;_--){
        cin>>n;cout<<P[n]<<endl;
    }
    return 0;
}
```

HDU4658 拆分时每个数的使用次数要小于  $k$

把生成函数改一下：

$$P(n)x^i = (1+x+x^2+\dots+x^{k-1})(1+x^2+x^4+\dots+x^{2(k-1)})\dots = \prod_{i=1}^{\infty} \frac{1-x^{ik}}{1-x^i}$$

```

#include<bits/stdc++.h>
#define ll long long
#define pii_ pair<int,int>
#define mp_ make_pair
#define pb push_back
#define fi first
#define se second
#define rep(i,a,b) for(int i=(a);i<=(b);i++)
#define per(i,a,b) for(int i=(a);i>=(b);i--)
#define show1(a) cout<<#a<<" = "<<a<<endl
#define show2(a,b) cout<<#a<<" = "<<a<<" "; cout<<#b<<" = "<<b<<endl
using namespace std;
const ll INF = 1LL<<60;
const int inf = 1<<30;
const int maxn = 2e5+5;
const ll M = 1e9+7;
inline void fastio() {ios::sync_with_stdio(false);cin.tie(0);cout.tie(0);}

ll f1[maxn],f2[maxn],P[maxn];
void init()
{
    int n = 1e5;
    for(int i=1;;i++){
        f1[i] = (3*i*i-i)>>1;
        if(f1[i]>n) break;
        f2[i] = (3*i*i+i)>>1;
        if(f2[i]>n) break;
    }
    P[0] = 1;
    rep(i,1,n){
        for(int j=1;;j++){
            if(f1[j]<=i) P[i] += (j&1)?P[i-f1[j]] : -P[i-f1[j]];
            else break;
            if(f2[j]<=i) P[i] += (j&1)?P[i-f2[j]] : -P[i-f2[j]];
            else break;
        }
        P[i] = (P[i]%M+M)%M;
    }
}

ll solve(int n,int k)
{
    ll ans = P[n];
    for(int i=1;;i++){
        ll a = (ll)k*(3*i*i-i)>>1;
        if(a>n) break;
        ll t = P[n-a];
        ll b = (ll)k*(3*i*i+i)>>1;
        if(b<=n) t = (t+P[n-b])%M;
        if(i&1) ans = (ans-t+M)%M;
        else ans = (ans+t)%M;
    }
}

```

```
    }
    return ans;
}
int main()
{
    fastio();
    init();
    int _,n,k;
    for(cin>>_;&_&_--){
        cin>>n>>k;cout<<solve(n,k)<<endl;
    }
    return 0;
}
```

## Infinity37

在搞期末考试，完全摸了

## Zars19

不好意思摸了摸了，我马上补555

## 本周推荐

From:  
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:  
[https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:wangzai\\_milk:weekly5&rev=1591530352](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:wangzai_milk:weekly5&rev=1591530352) 

Last update: **2020/06/07 19:45**