# 【模板】多项式除法

2018年07月29日 11:27:54 [ccosi](https://me.csdn.net/corsica6" \t "https://blog.csdn.net/corsica6/article/details/_blank) 阅读数：1807

 版权声明：欢迎转载(请附带原链接)ヾ(๑╹◡╹)ﾉ https://blog.csdn.net/corsica6/article/details/81269393

传送门:[洛谷：【模板】多项式除法](https://www.luogu.org/problemnew/show/P4512" \t "https://blog.csdn.net/corsica6/article/details/_blank)

# **题意**

给定一个 n 次多项式 F(x) 和一个 m 次多项式 G(x) ，请求出多项式 Q(x) ,R(x) ，满足以下条件：   
Q(x) 次数为 n−m，R(x) 次数小于 m   
F(x)=Q(x)×G(x)+R(x)   
所有的运算在模 998244353 意义下进行。

# **题解**

由F(x)=Q(x)×G(x)+R(x) (mod xn+1)   
得F(1/x)=Q(1/x)×G(1/x)+R(1/x) (mod xn+1)   
同乘上一个xn即Frev(x)=Qrev(x)×Grev(x)+Rrev(x)xn−degR (mod xn+1)   
其中Frev(x)表示将多项式F(x)的0,1...n次项的系数翻转。degR表示R的次数。   
因为degR<m，显然n−degR>n−m。而degQ=n−m   
那么在mod n−m+1意义下，存在：   
Frev(x)=Qrev(x)×Grev(x) (mod xn−m+1)   
Rrev既然已被消去，那么只需要一个多项式求逆即可得出Qrev(x)，在翻转回去就是答案，而degQ<n−m+1，所以答案满足条件，成立。   
得出Q(x)后，直接带回在(mod xn+1)意义下做个多项式乘法，用F(x)减去就是R(x)。

# **代码**

#include<bits/stdc++.h>

using namespace std;

#define ll long long

#define re register

#define gc get\_char

#define cs const

#define cerr fout

namespace IO{

inline char get\_char(){

static cs int Rlen=1<<20|1;

static char buf[Rlen],\*p1,\*p2;

return (p1==p2)&&(p2=(p1=buf)+fread(buf,1,Rlen,stdin),p1==p2)?EOF:\*p1++;

}

inline int getint(){

re char c;

while(!isdigit(c=gc()));re int num=c^48;

while(isdigit(c=gc()))num=(num+(num<<2)<<1)+(c^48);

return num;

}

}

using namespace IO;

cs int mod=998244353;

inline int add(cs int &a,cs int &b){return a+b>=mod?a+b-mod:a+b;}

inline int dec(cs int &a,cs int &b){return a<b?a-b+mod:a-b;}

inline int mul(cs int &a,cs int &b){return (ll)a\*b%mod;}

inline int quickpow(int a,int b,int res=1){

while(b){

if(b&1)res=mul(res,a);

a=mul(a,a);

b>>=1;

}

return res;

}

cs int N=1<<20|1;

int r[N];

inline void NTT(int \*A,int len,int typ){

for(int re i=0;i<len;++i)if(i<r[i])swap(A[i],A[r[i]]);

for(int re i=1;i<len;i<<=1){

int wn=quickpow(typ==-1?(mod+1)/3:3,(mod-1)/i/2);

for(int re j=0;j<len;j+=i<<1)

for(int re k=0,x,y,w=1;k<i;++k,w=mul(w,wn)){

x=A[j+k],y=mul(w,A[j+k+i]);

A[j+k]=add(x,y);

A[j+k+i]=dec(x,y);

}

}

if(typ==-1)

for(int re i=0,inv=quickpow(len,mod-2);i<len;++i)

A[i]=mul(A[i],inv);

}

inline void Inv(int deg,int \*a,int \*b){

static int c[N];

if(deg==1){b[0]=quickpow(a[0],mod-2);return ;}

Inv((deg+1)>>1,a,b);

int re len=1;

while(len<(deg<<1))len<<=1;

for(int re i=0;i<len;++i)r[i]=(r[i>>1]>>1)|((i&1)\*(len>>1));

memcpy(c,a,sizeof(int)\*deg);

memset(c+deg,0,sizeof(int)\*(len-deg));

NTT(c,len,1),NTT(b,len,1);

for(int re i=0;i<len;++i)b[i]=mul(b[i],dec(2,mul(b[i],c[i])));

NTT(b,len,-1);

memset(b+deg,0,sizeof(int)\*(len-deg));

}

inline void mul(int \*A,int \*B,int deg){

static int a[N],b[N];

int re len=1;

while(len<(deg<<1))len<<=1;

for(int re i=0;i<len;++i)r[i]=r[i>>1]>>1|((i&1)\*(len>>1));

memcpy(a,A,sizeof(int)\*deg),memcpy(b,B,sizeof(int)\*deg);

memset(a+deg,0,sizeof(int)\*(len-deg)),memset(b+deg,0,sizeof(int)\*deg);

NTT(a,len,1),NTT(b,len,1);

for(int re i=0;i<len;++i)a[i]=mul(a[i],b[i]);

NTT(a,len,-1);

memcpy(A,a,sizeof(int)\*deg);

}

int n,m,F[N],G[N],Gr[N],R[N],Q[N];

signed main(){

n=getint()+1,m=getint()+1;

for(int re i=0;i<n;++i)Q[n-1-i]=F[i]=getint();

for(int re i=0;i<m;++i)Gr[m-1-i]=G[i]=getint();

Inv(n-m+1,Gr,R);//求G(x)^-1

mul(Q,R,n-m+1);

reverse(Q,Q+n-m+1);//Q(x)=F(x)\*G(x)^-1

for(int re i=0;i<n-m+1;++i)cout<<Q[i]<<" ";cout<<"\n";

for(int re i=n-m+1;i<=n;++i)Q[i]=0;

mul(Q,G,n<<1);

for(int re i=0;i<m-1;++i)cout<<dec(F[i],Q[i])<<" ";//R(x)=F(x)-G(x)\*Q(x)

return 0;

}