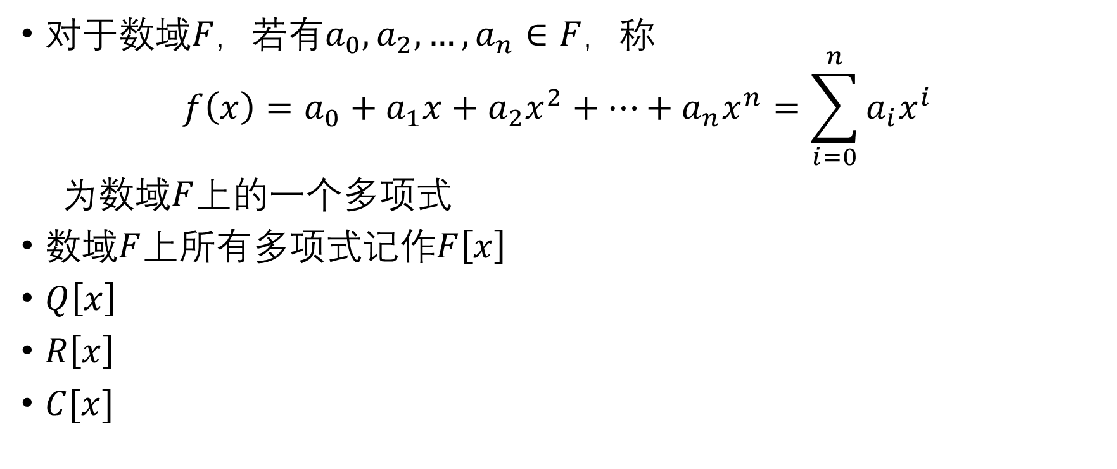
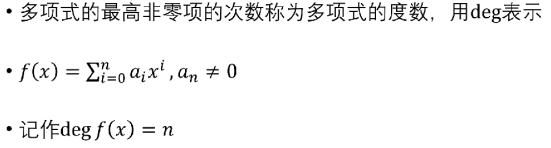
# **快速傅里叶变换（FFT）详解**

快速傅里叶变换(Fast Fourier Transformation)简称FFT。在各大OI竞赛中也常有用到，也是一个十分优秀的~~可以装逼的~~好算法

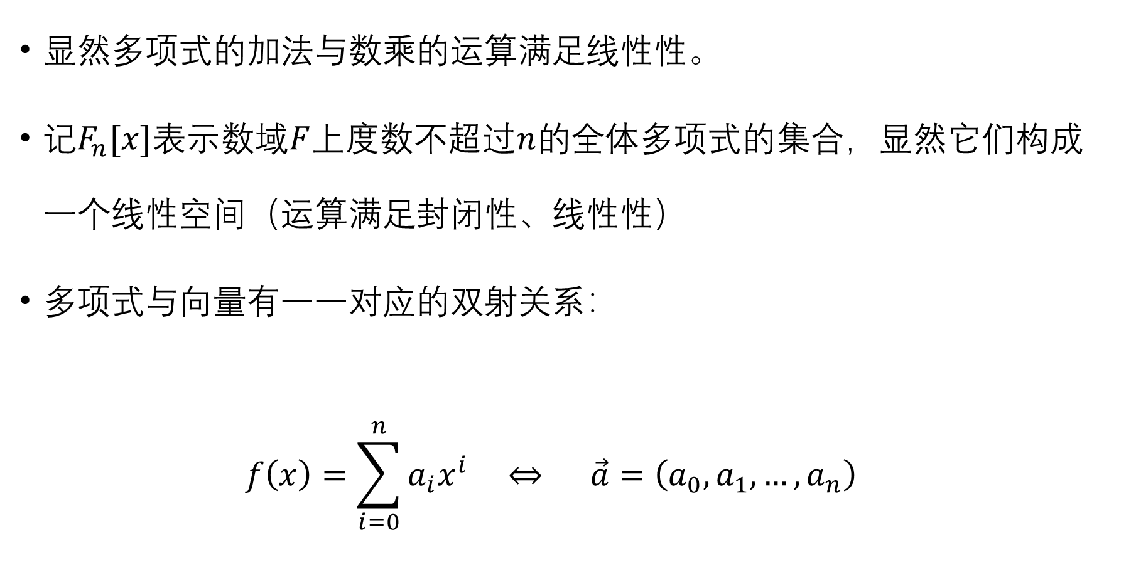
### **基础知识：**



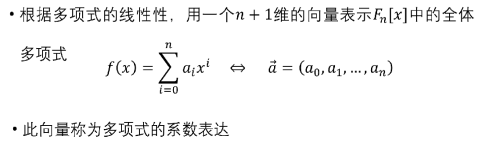
### **多项式的度数：**



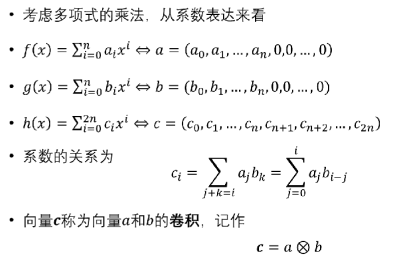
### **多项式的线性空间**



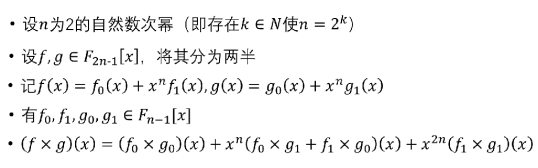
### **系数表达**

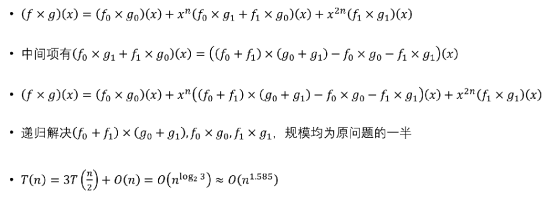


### **向量的卷积**

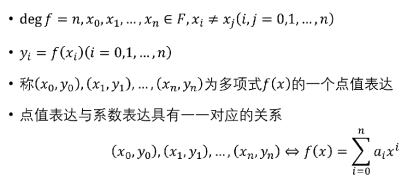


### **分治乘法**



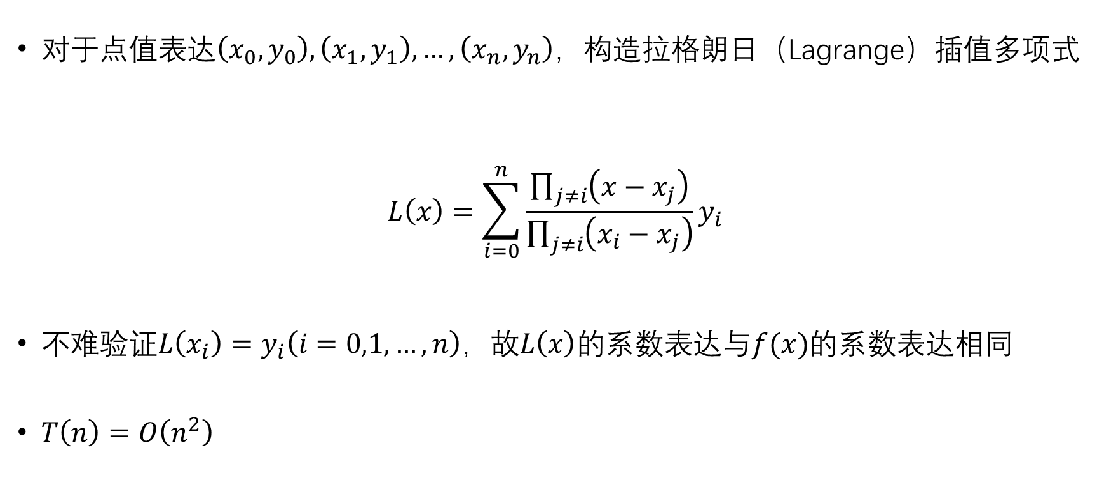


### **点值表达**

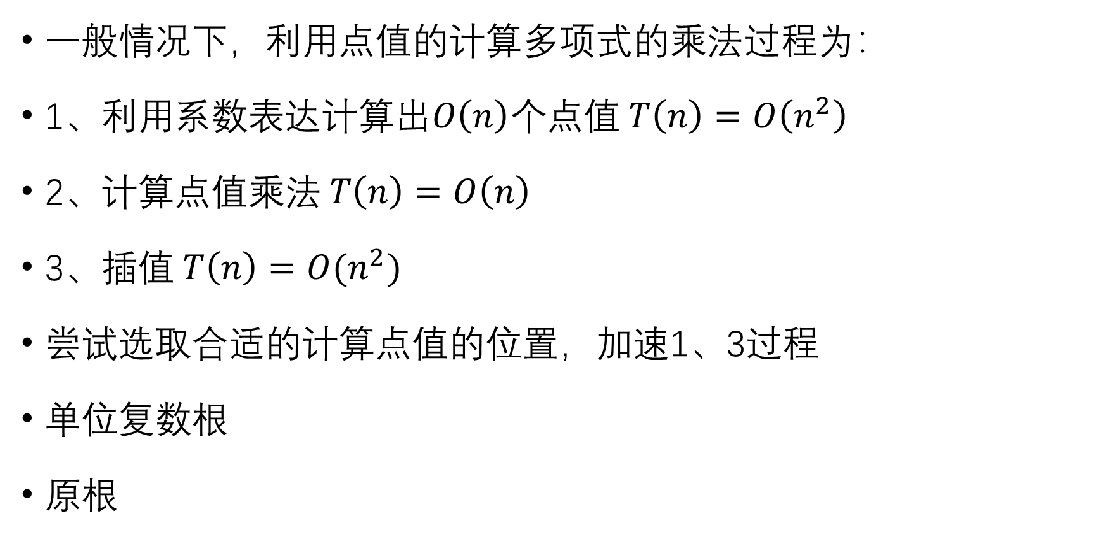


### **插值**

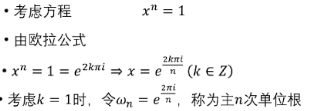
### 

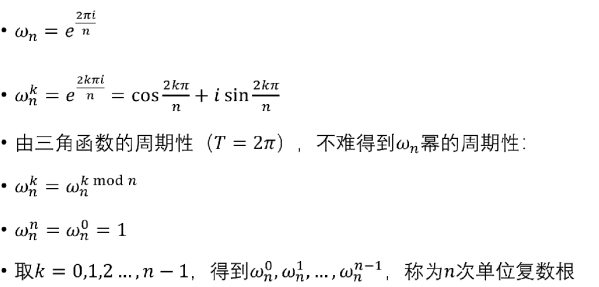
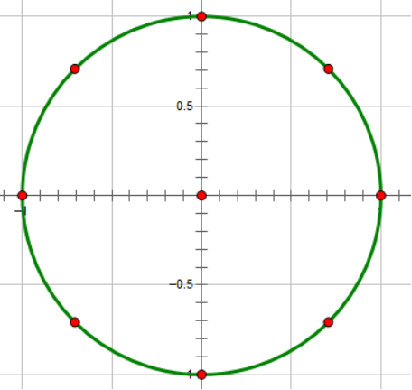


### **点值计算分析**



### **单位复数根**

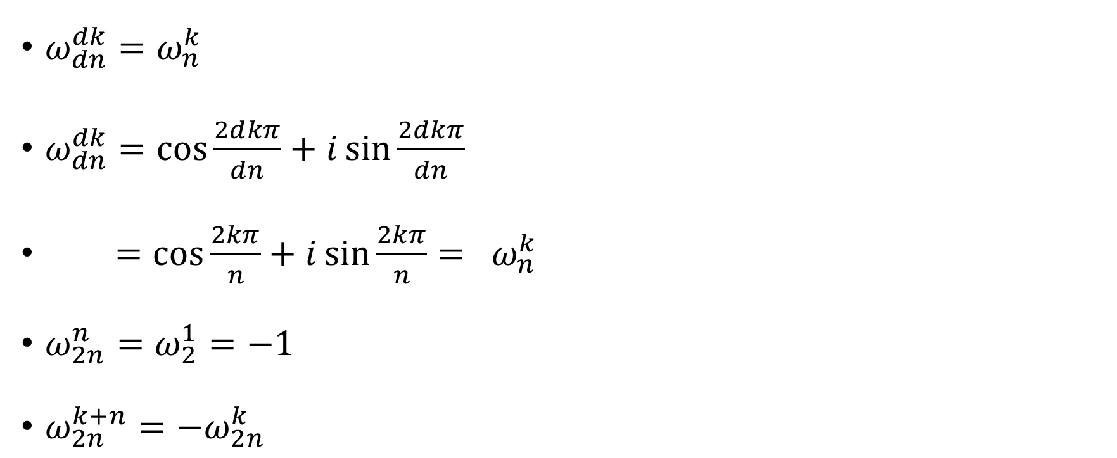


****

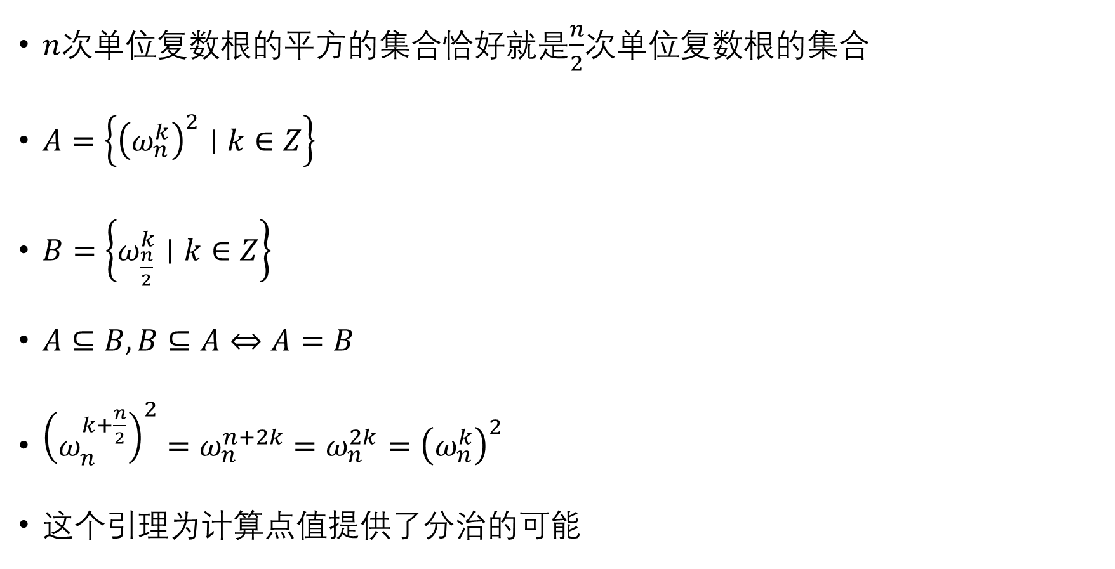
### 

### **单位复数根的性质**

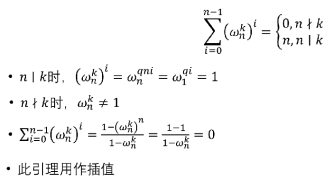
1.消去引理



　　2.折半引理

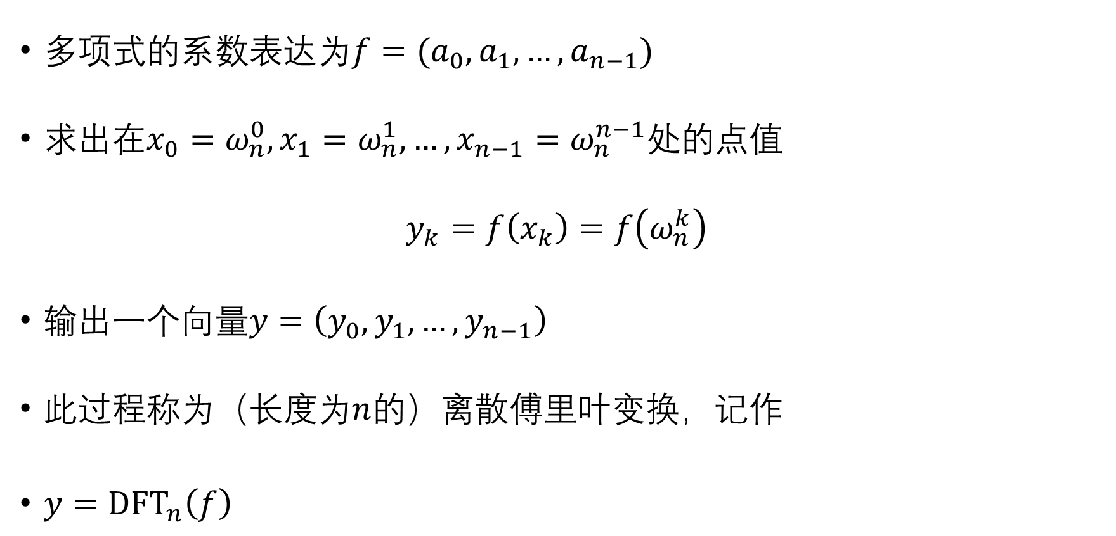


　　3.求和引理

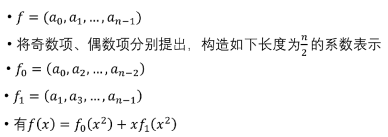
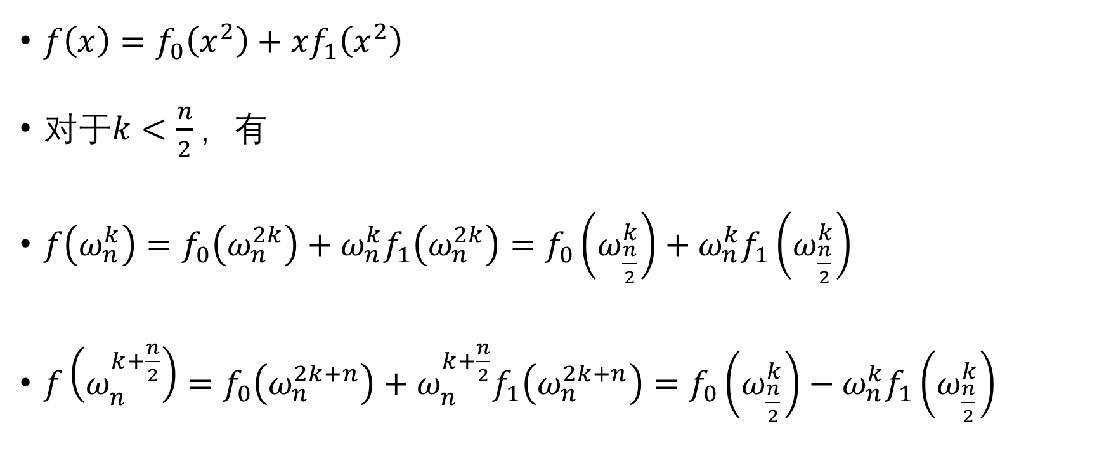


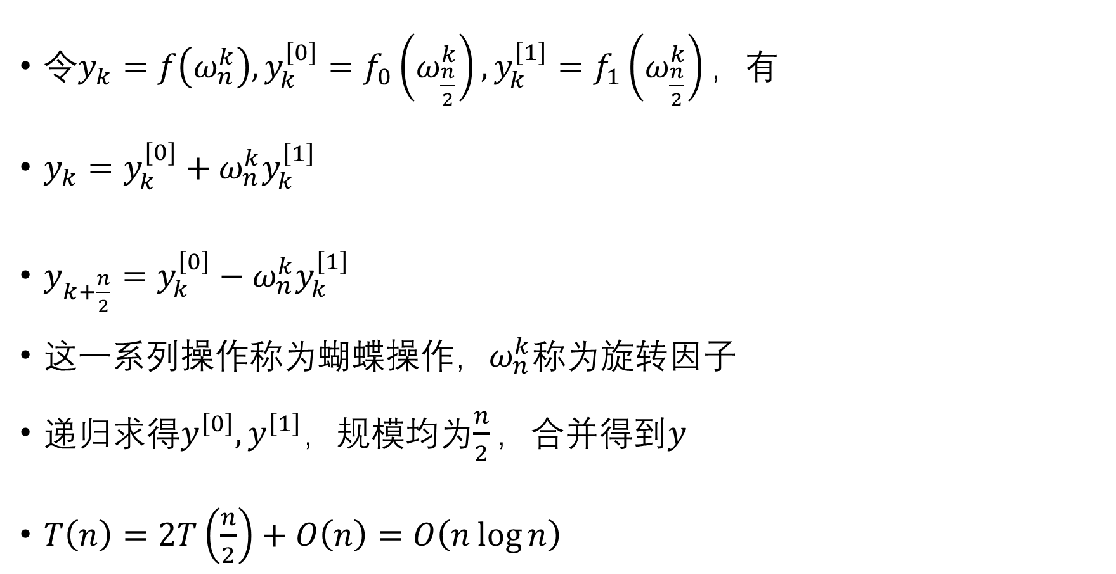
### **铺垫都铺完了，让我们一起进入DFT,FFT,IDFT的~~美妙~~世界吧！**

### **离散傅里叶变换（Discrete Fourier Transform 简称DFT）**

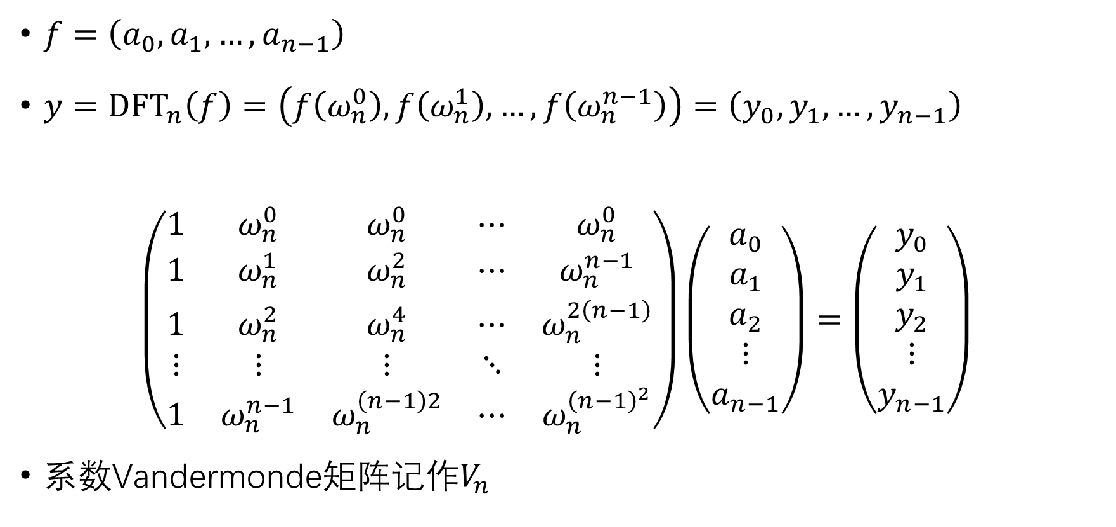


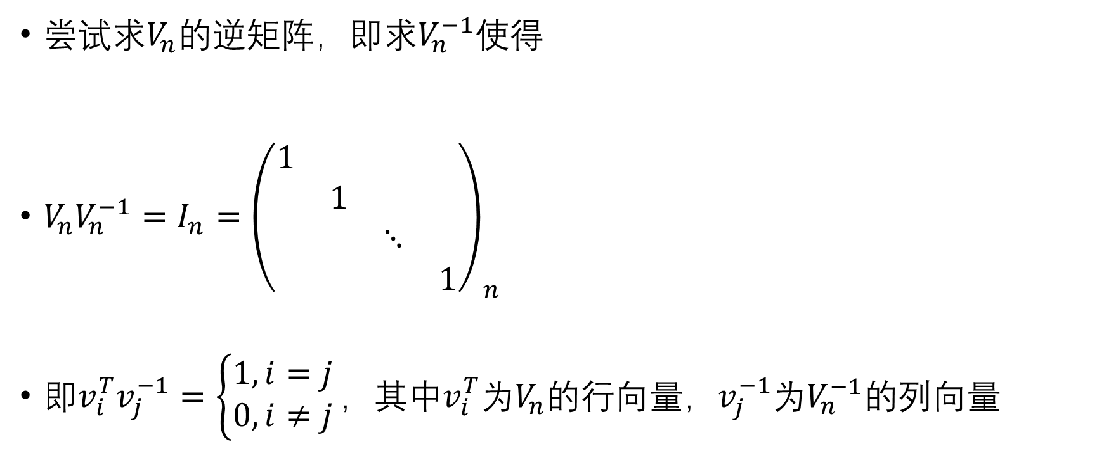
### **快速傅里叶变换（FFT）**

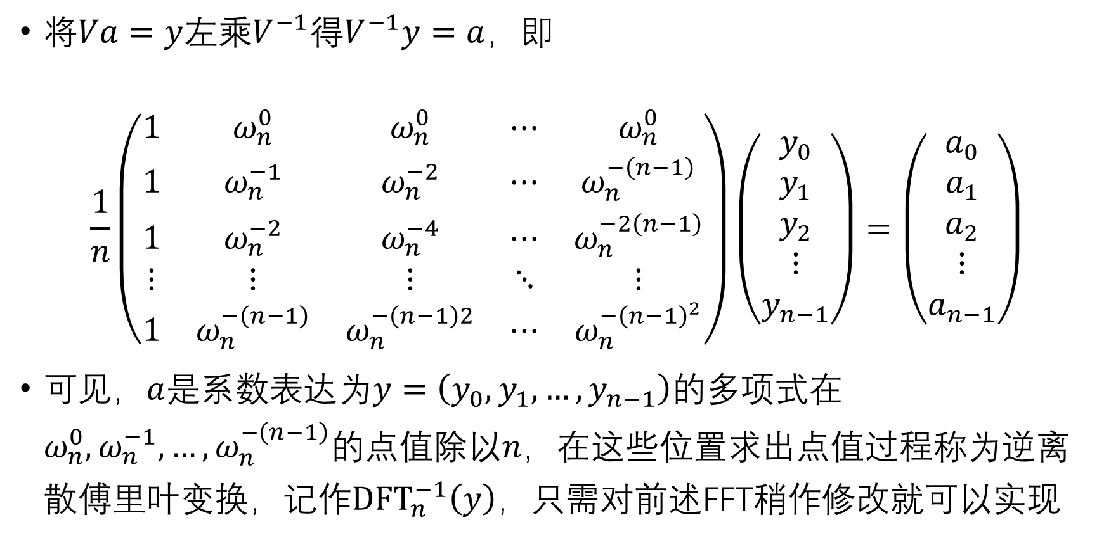
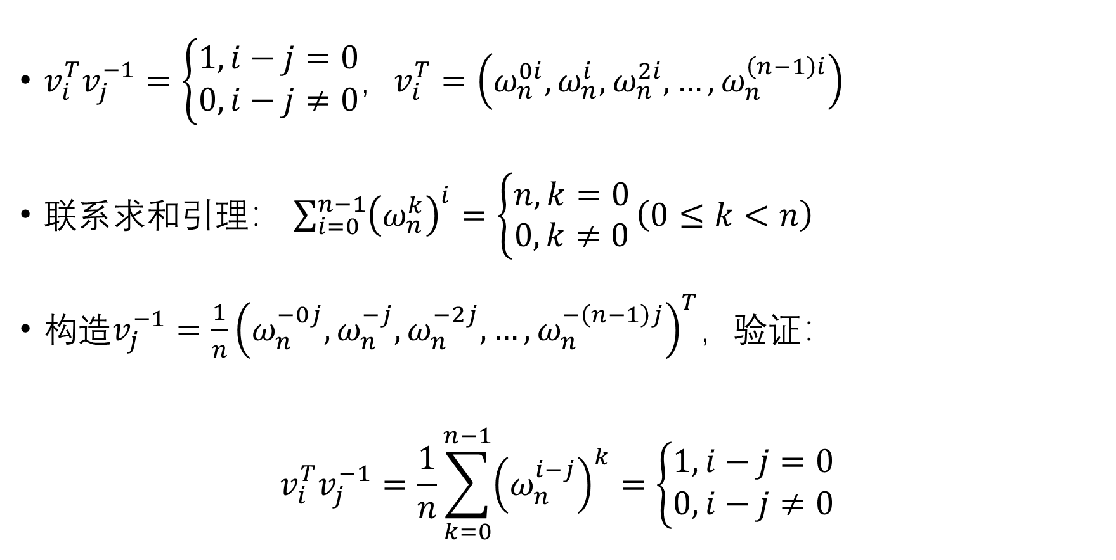
~~~~



### **逆离散傅里叶变换（Inverse Discrete Fourier Transform 简称IDFT）**

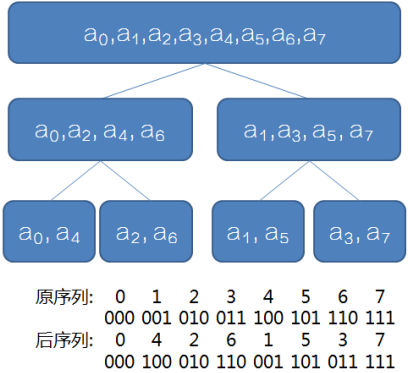


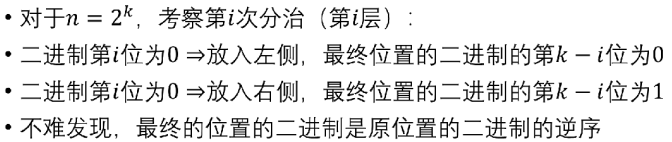




### **FFT的迭代实现**

我们类似于需要像这样实现FFT：





### **知识点终于讲完了，接下来我们就要开始写板子了**

板子题：[uoj #34](http://uoj.ac/problem/34" \t "https://www.cnblogs.com/wangyh1008/p/_blank)

代码附上~~

#include<cstdio>

#include<iostream>

#include<cmath>

#include<cstring>

#include<algorithm>

#include<cstdlib>

using namespace std;

const int mod=1e9+7;

const double pi=acos(-1);

struct cn

{

double x,y;

cn (double x=0,double y=0):x(x),y(y) {}

}a[300005],b[300005],c[300005];

cn operator + (const cn &a,const cn &b) {return cn(a.x+b.x,a.y+b.y);}

cn operator - (const cn &a,const cn &b) {return cn(a.x-b.x,a.y-b.y);}

cn operator \* (const cn &a,const cn &b) {return cn(a.x\*b.x-a.y\*b.y,a.x\*b.y+a.y\*b.x);}

void fft(cn a[],int n,int l,int f){

int rev[n+5];

rev[0]=0;

//这一步就是二进制的转置部分,r[i]表示i的二进制转置,比如说r[6(110)]=3(011)

//r[i]由r[i/2]递推得来,对比i和i/2的二进制规律,我们发现i=(i>>2)<<1+(i&1)

//因为r[i]是i的倒序,所以也应该是倒序递推

for (int i=1; i<n; i++){//按照递归，调整数据次序

rev[i]=(rev[i>>1]>>1)|((i&1)<<l-1);

 if (i<rev[i]) swap(a[i],a[rev[i]]);

}

for (int i=1; i<n; i<<=1){//倍增次数

cn wi(cos(pi/i),f\*sin(pi/i));//x

for (int j=0; j<n; j+=i\*2){//j，枚举计算函数f的第1个数

cn w(1,0);

for (int k=0; k<i; k++){//k,计算函数f的个数

cn x=a[j+k],y=w\*a[j+k+i];

a[j+k]=x+y;

a[j+k+i]=x-y;

w=w\*wi;//x^2,x^3,……

}

}

}

if (f==-1)

for (int i=0; i<n; i++){

a[i].x/=n; a[i].y/=n;

}

}

int main()

{

int n,m;

scanf("%d%d",&n,&m); n++; m++;

for (int i=0; i<n; i++) scanf("%lf",&a[i].x);

for (int i=0; i<m; i++) scanf("%lf",&b[i].x);

int l=0,N=1;

while (N<n+m-1) N<<=1,l++;

fft(a,N,l,1);

fft(b,N,l,1);

for (int i=0; i<N; i++) c[i]=a[i]\*b[i];

fft(c,N,l,-1);

for (int i=0; i<n+m-1; i++) printf("%d ",(int)(c[i].x+0.5));

return 0;

}

