利用FFT计算多项式相乘

# 卷积的典型应用

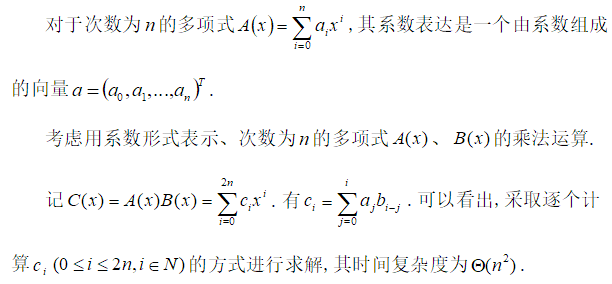
卷积的最典型的应用就是多项式乘法（多项式乘法就是求卷积）。

卷积可以通过FFT算法快速实现。

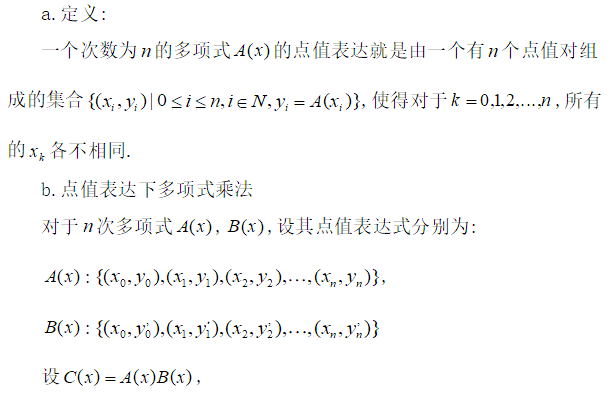
(请参见：利用FFT实现卷积)

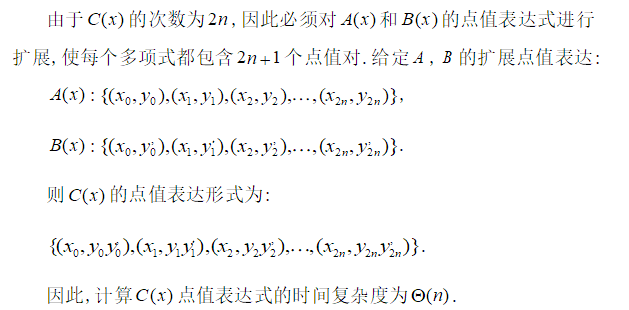
多项式的表达式：

## 系数表达式：



## 点值表达





# Matlab实现

方法1：直接利用conv(A,B)即可实现；

方法2：手写：利用FFT实现循环卷积

function [result] = polyMultiByFFT(A,B)

% A first polynomial ,high first,low after

% A first polynomial ,high first,low after

% result ,high bit first,low bit after

% the result is same as conv(A,B).

N\_A = length(A);

N\_B = length(B);

N\_use = N\_A+N\_B-1;

N\_R = pow2(ceil(log2(N\_use)));

A\_new = [A,zeros(1,N\_R-N\_A)];

B\_new = [B,zeros(1,N\_R-N\_B)];

Y\_A = fft(A\_new);

Y\_B = fft(B\_new);

Y\_R = Y\_A.\*Y\_B;

y\_r = ifft(Y\_R);

result = y\_r(1:N\_use);

end

测试函数：

clc,clear

A = [1 0 1];%A多项式

B = [1 0 1];%B多项式

result = polyMultiByFFT(A,B);

result%1 0 2 0 1

因为FFT是点数对应的是2的幂，所以当N\_A+N\_B-1不足2的幂的时候，取最靠近的2的幂的值；最后把多余的零删去。