实验四题目：

设计计算机程序，产生序列并计算序列的FFT和IFFT，绘制其幅频特性和相频特性曲线；

模拟产生离散系统的输入序列和单位脉冲响应，利用FFT和IFFT算法计算系统的输出响应，分析FFT的计算长度对系统输出响应的影响；

模拟产生连续时间信号，选取适当的采样频率对其采样，并用FFT算法计算其频谱，分析信号的观测时间长度、FFT的计算长度对信号频谱计算结果的影响。

1. 设计计算机程序，产生序列并计算序列的FFT和IFFT，绘制其幅频特性和相频特性曲线。

产生序列为

x(n)=[1,4,3,6,5]

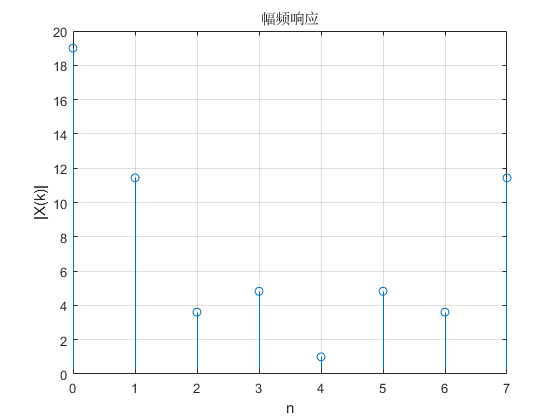
设N=8，则计算该序列的八点FFT为

|  |  |
| --- | --- |
| k | X（k） |
| 0 | 19.0000000000000 + 0.00000000000000i |
| 1 | -5.41421356237310 - 10.0710678118655i |
| 2 | 3.00000000000000 + 2.00000000000000i |
| 3 | -2.58578643762691 - 4.07106781186548i |
| 4 | -1.00000000000000 + 0.00000000000000i |
| 5 | -2.58578643762691 + 4.07106781186548i |
| 6 | 3.00000000000000 - 2.00000000000000i |
| 7 | -5.41421356237310 + 10.0710678118655i |

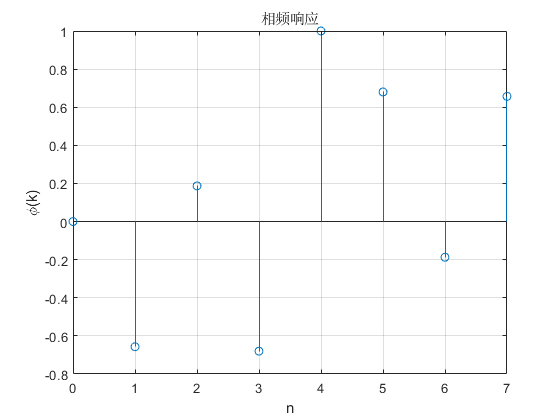
该序列的IFFT为

x(n)=[1,4,3,6,5,0,0,0]

幅频特性如图所示：



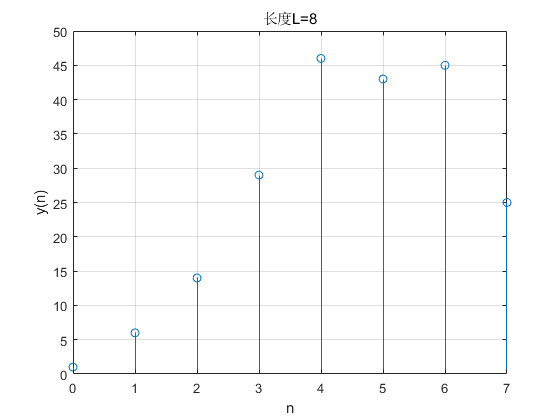
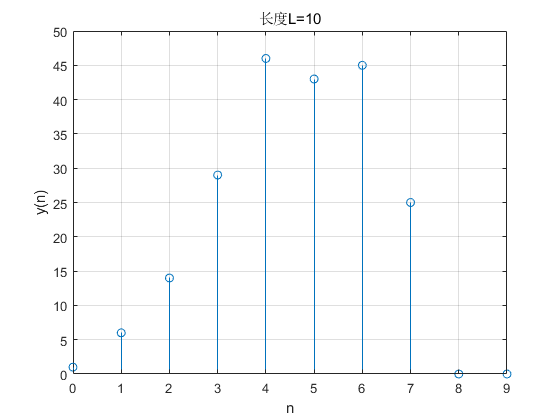
相频特性如图所示：

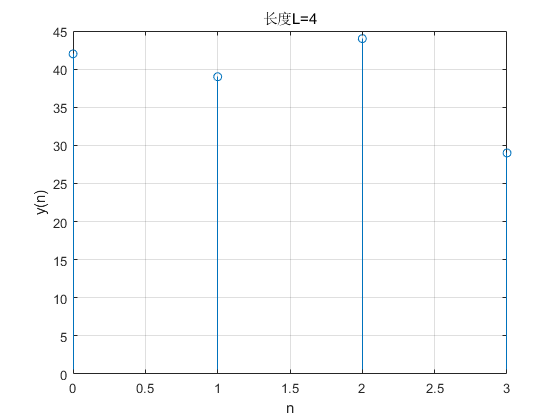
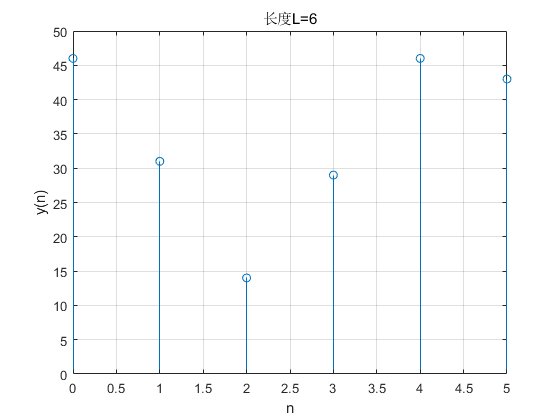


1. 模拟产生离散系统的输入序列和单位脉冲响应，利用FFT和IFFT算法计算系统的输出响应，分析FFT的计算长度对系统输出响应的影响。

设输入序列为x(n)=[1,4,3,6,5]，单位脉冲响应为h(n)=[1,2,3,5]。

分别取FFT的计算长度为10，8，6，4，计算系统输出响应如下图所示：





从图中可以看出，当长度L=10和L=8时，系统输出响应y(n)在[0,7]上完全相等。

当L=6时，n取2,3,4,5的y(n)没有变化，但y(0)和y(1)的值却异常增大。通过计算可以发现，此时的y(0)为L取8时的y(0)与y(6)之和，同样，此时的y(1)为L取8时的y(1)与y(7)之和。

考虑到x(n)的长度为5<8，因此对其做FFT则最小需要的计算长度为8，即三级蝶形运算。而当L取6时，受到循环卷积的影响，原本的y(6),y(7)和y(0),y(1)发生混叠，从而导致此类图像。

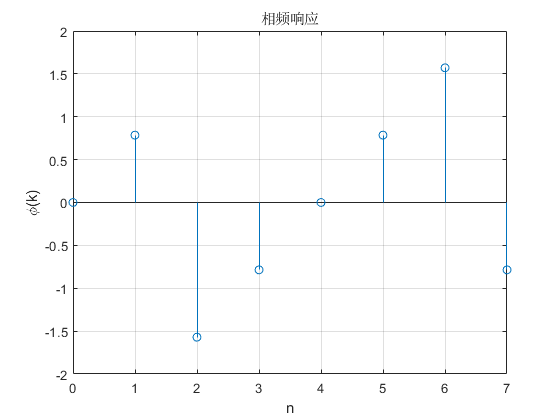
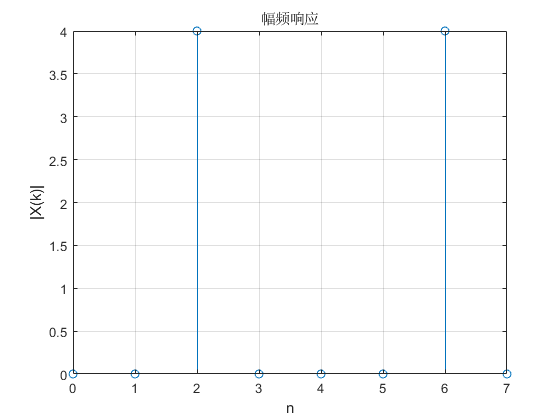
3、模拟产生连续时间信号，选取适当的采样频率对其采样，并用FFT算法计算其频谱，分析信号的观测时间长度、FFT的计算长度对信号频谱计算结果的影响。

设模拟信号为 。

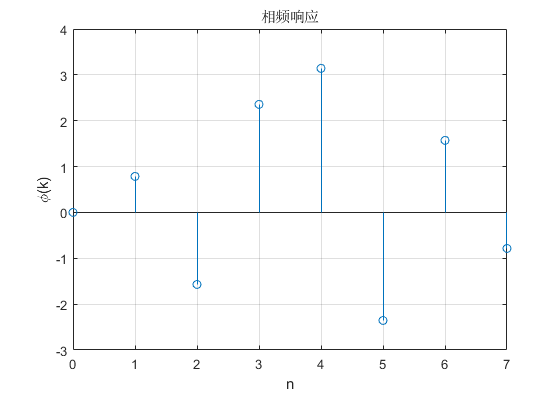
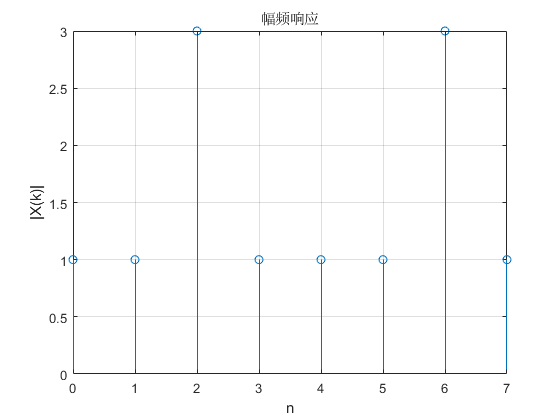
3.1 观测时间长度对信号频谱计算结果的影响

观测时间长度即采样信号的长度，为了探究其对信号频谱计算结果的影响，我们固定FFT的计算长度为8，分别取采样信号长度为8，6，4，绘制信号频谱图如下所示：

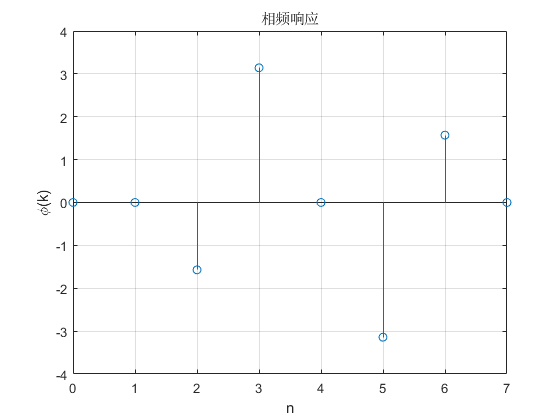
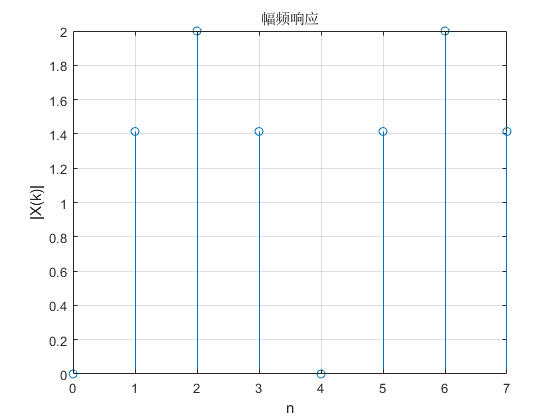
采样信号长度=8



采样信号长度=6



采样信号长度=4

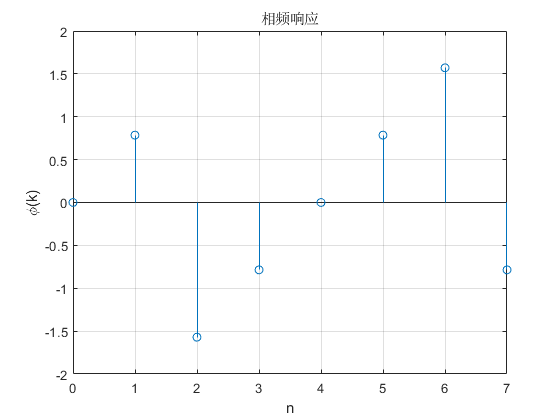
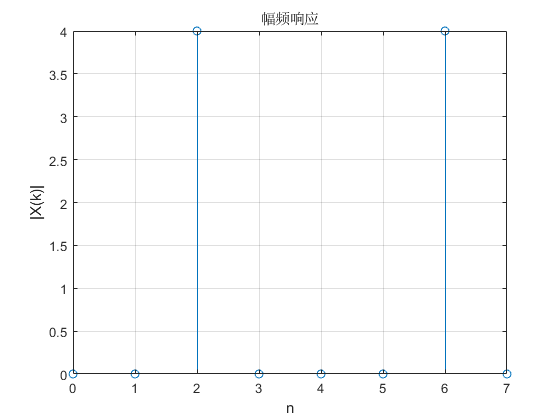


从图中我们可以发现，当FFT的计算长度等于序列长度时，信号的幅频响应仅采集到原序列的波峰位置。当FFT的计算长度逐渐大于序列长度时，信号越趋近于真实值。

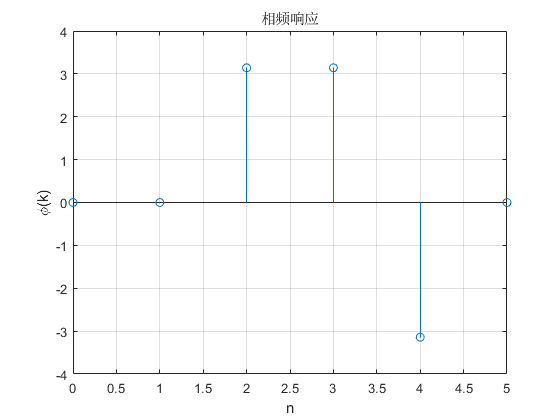
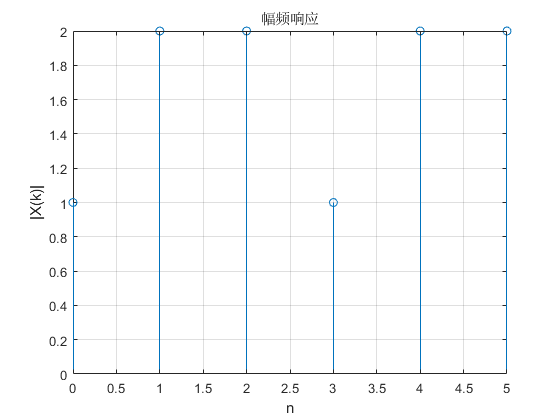
3.2 FFT的计算长度对信号频谱计算结果的影响

为了探究FFT的计算长度对信号频谱计算结果的影响，我们固定观测时间长度为8，分别取FFT的计算长度为8，6，4，绘制信号频谱图如下所示：

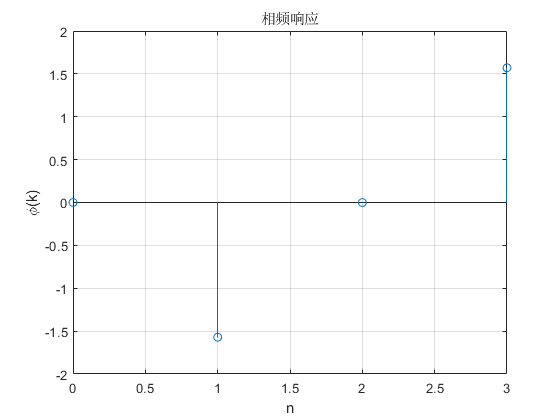
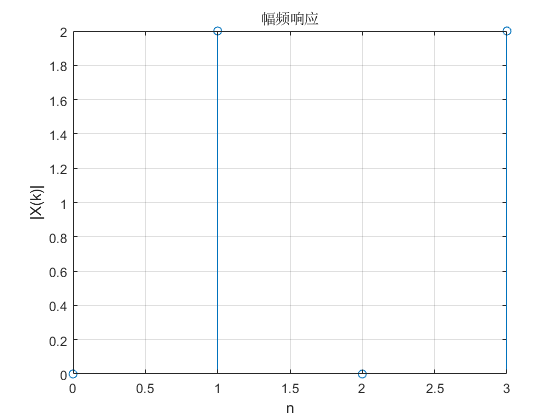
FFT的计算长度=8



FFT的计算长度=6



FFT的计算长度=4

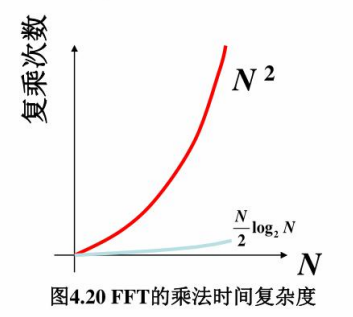


从图中我们可以发现，当FFT的计算长度等于序列长度时，信号频谱正常计算。当FFT的计算长度小于序列长度时，幅频和相频均发生混叠，原因同问题二。

问题的提出与解决：

章星宇19200300029

问题：FFT是对DFT算法的优化。上课中提到DFT算法的时间复杂度为，而FFT算法的时间复杂度为，两者复杂的复乘次数随N的变化如下图所示。在matlab上是否能通过仿真来验证这一变化规律？



解决：

Matlab内置了ftt函数，而dft的函数需要自行编写，仿真代码见附录。在仿真中，对N分别取10，100，1000，10000，计算所需时间，结果如下表所示。

|  |  |  |
| --- | --- | --- |
| N | FFT所需时间(秒） | DFT所需时间(秒） |
| 10 | 0.0031 | 0.0054 |
| 100 | 0.0032 | 0.0056 |
| 1000 | 0.0036 | 0.1323 |
| 10000 | 0.0047 | 19.62 |

可以发现，当N取值为100以内时，两者差别不大，而当N取值为10000时，DFT需要消耗19.62秒，而FFT的时间消耗仍小于0.01秒，从而验证了当N取值越大时两者的差异越明显。

程序代码：

1. % 问题一
2. x=[1,4,3,6,5];
3. N=8;
4. X=fft(x, N);
5. x=ifft(X, N);
6. figure(1)
7. n=0:1:7;
8. stem(n,abs(X));
9. title('幅频响应');
10. xlabel('n'),ylabel('|X(k)|');
11. grid on;
13. figure(2)
14. stem(n,angle(X)/pi);
15. title('相频响应');
16. xlabel('n'),ylabel('\phi(k)');
17. grid on;

20. % 问题二
21. h=[1,2,3,5];
22. x=[1,4,3,6,5];
24. % 此处设置不同的L
25. L=4;
26. y=ifft(fft(h,L).\*fft(x,L));
27. i=0:1:L-1;
28. figure(3)
29. stem(i,y);
30. xlabel('n')
31. ylabel('y(n)')
32. title('长度L=4');
33. grid on;

36. % 问题三
37. n = 0:1:7
38. L=8;
39. f=sin(n \* pi \*0.5);
40. F=fft(f,L);
41. i=0:1:L-1;
42. figure(4)
43. stem(i,abs(F))
44. title('幅频响应');
45. grid on;
46. xlabel('n'),ylabel('|X(k)|');
47. figure(5)
48. stem(i,angle(F))
49. title('相频响应');
50. xlabel('n'),ylabel('\phi(k)');
51. grid on;
53. % 计算DFT和FFT消耗时间(对n分别取10，100，1000，10000)
54. tic
55. n = 0:1:10;
56. f=sin(n \* pi \*0.5);
57. F=fft(f);
58. toc
60. tic
61. n = 0:1:10;
62. L=8;
63. f=sin(n \* pi \*0.5);
64. F=dft(f);
65. toc
67. %自定义DFT函数
68. function xk=dft(xn)
69. N=length(xn);
70. WN=exp(-j\*2\*pi/N);
71. n=0:1:N-1;     %定义一个一维矩阵，即行向量,从0到N-1
72. k=0:1:N-1;
73. nk=k'\*n;      %行向量k变换为列向量 乘上 行向量n ，得到一个N x N的矩阵
74. WNnk=WN.^(nk);      %做幂运算后的参数仍为一个 N x N的系数矩阵
75. xk=xn\*WNnk;       %行向量 乘以  N x N的系数矩阵 即为DFT变换后的矩阵
76. end