**实验一 常见离散信号产生和实现**

**Part01 Examples in Chapter 2**

1. Program\_2\_1

（1）功能：产生数据s[n]和噪声d[n]，模拟集合平均去噪的过程。

（2）运行结果：



（3）思考：测量总的次数对去噪效果有什么影响？

（4）实验：

测量次数n=5，运行结果： 测量次数n=100，运行结果：

 

（5）结论：

测量总的次数越多，去噪效果效果越好。

2. Program\_2\_2

（1）功能：分别输入两个有限长序列a和b，计算a和b的卷积。

（2）运行结果：

输入：a=[-2 0 1 -1 3] b=[1 2 0 -1 0]

输出：output sequence =

-2 -4 1 3 1 5 1 -3 0



（3）思考：如何计算有限长连续函数的卷积？

（4）实验：

代码：

clear;

x = -1 :0.01: 1;

fx = x; %函数f(x)=x,自变量范围 -1<=x<=1

x = 0 :0.01: 1;

gx = x+1; %函数g(x)=x+1, 自变量范围 0<=x<=1

yx = conv(fx, gx); % 进行卷积计算

x=-1:0.01:2; %卷积结果变量范围为（原函数自变量最小值）~（两原函数

自变量最大值相加）

plot(x,yx);

结果：



3. Program\_2\_3

（1）功能：输入一个复指数序列的各个参数，分别输出这个复指数序列的实部和虚部的散点图。

（2）实验结果：

输入各参数：x[n]=exp((-1/12+jπ/6)n)

输出：



（3）思考：复指数的实部a和虚部b对整个函数实部和虚部的影响

（4）实验：

x[n]=exp((-1/3+jπ/6)n)：



x[n]=exp((-1/24+jπ/6)n)：



x[n]=exp((-1/12+jπ/4)n)



x[n]=exp((-1/12+jπ/12)n)



（5）结论：

复指数的实部的绝对值越大，震荡消失得越快。

复指数的虚部则影响震荡周期的大小。

4. Program\_2\_4

（1）功能：输入一个实指数序列的参数，输出这个实指数序列的图像。

（2）实验结果：

输入：x=0.2\*(1.2)^n

输出：



（3）思考：底数a对实指数函数的影响

（4）实验：

x=0.2\*(1.3)^n x=0.2\*(-1.2)^n

 

x=0.2\*(0.8)^n x=0.2\*(0.6)^n

 

（5）结论：

记底数为a，则 |a-1| 越大，函数值变化越快；若a<0，则函数值正负交替变化。

5. Program\_2\_5

（1）功能：输入两个有限长序列x和y，计算这两个有限长序列的互相关序列。

（2）实验结果：

输入：[1,3,-2,1,2,-1,4,4,2] 和 [2,-1,4,1,-2,3]

输出：



（3）思考：验证自相关和互相关序列的性质 。

（4）实验：

代码：

clear;

clc;

x = [1,3,-2,1,2,-1,4,4,2];

y = [2,-1,4,1,-2,3];

% Compute the correlation sequence

n1 = length(y)-1; n2 = length(x)-1;

r = conv(x,fliplr(y));

r1=xcorr(x)

r2=xcorr(y)

k = (-n1):n2;

subplot(131);stem(k,r);xlabel('xy');

k = (-n2):n2;

subplot(132);stem(k,r1);xlabel('x');

k = (-n1):n1;

subplot(133);stem(k,r2);xlabel('y');

输出结果：

r1 =

Columns 1 through 14

2.0000 10.0000 12.0000 5.0000 -1.0000 19.0000 8.0000 15.0000 56.0000 15.0000 8.0000 19.0000 -1.0000 5.0000

Columns 15 through 17

12.0000 10.0000 2.0000

r2 =

6.0000 -7.0000 16.0000 2.0000 -10.0000 35.0000 -10.0000 2.0000 16.0000 -7.0000 6.0000



（5）结论：

=56，=35，=44.27，性质成立。

6. Program\_2\_6

（1）功能：模拟一个受噪声干扰的周期序列，通过求这个序列的自相关序列来观察周期函数的周期。

（2）实验结果：



通过观察上图可以知道周期为8。

（3）思考：为什么要用自相关序列而不直接观察加噪后的序列？

（4）实验：

代码：

clear;

clc;

N = 96;

n = 1:N;

x = cos(pi\*0.25\*n); % Generate the sinusoidal sequence

d = rand(1,N) - 0.5; % Generate the noise sequence

y = x + d; % Generate the noise-corrupted sinusoidal sequence

k=1:56;

subplot(121);stem(k,y(1:56));

xlabel('Lag index'); ylabel('Amplitude');

r = conv(y, fliplr(y)); % Compute the correlation sequence

k = -28:28;

subplot(122);stem(k, r(68:124));

xlabel('Lag index'); ylabel('Amplitude');

输出：



（5）结论：

直接绘制出加噪后的序列虽然也能大概看出具有周期性，但无法知道准确的周期，而因为噪声序列样本之间互不相关，所以进行过自相关处理的序列可以大大消去噪声序列的影响进而很好得体现出原周期函数的性质，所以能比较精确的看出其周期。

**Part02 Matlab Exercises in Chapter 2**

1. M2.1

（1）问题或改进：直接绘制xcs和xca的图像只能得到其实部的图像，并不能验证例2.8一定成立。

（2）修改后代码：

a=input('Type in real part =');

b=input('Type in imaginary part =');

% c=a+b\*i;

n=0:length(a)-1;

%x=K\*exp(c\*n);

x=a+b\*1i;

xcs=0.5\*(x+conj(fliplr(x)));%conjugate symmetric part of sequence

xca=0.5\*(x-conj(fliplr(x)));%conjugate antisymmetric part of sequence

figure,

subplot(2,2,1);stem(n,real(xcs)); %共轭对称部分的实部

xlabel('n');ylabel('real\_xca[n]');

subplot(2,2,2);stem(n,imag(xcs)); %共轭对称部分的虚部

xlabel('n');ylabel('imag\_xca[n]');

subplot(2,2,3);stem(n,real(xca)); %共轭反对称部分的实部

xlabel('n');ylabel('real\_xcs[n]');

subplot(2,2,4);stem(n,imag(xca)); %共轭反对称部分的虚部

xlabel('n');ylabel('imag\_xcs[n]');

（3）运行结果：

输入： Type in real part =[0 1 -2 4 -5 0 3]

Type in imaginary part =[0 4 3 -2 -6 -2 0]

输出：



由返回的图像可以验证例2.8

2. M2.2

（1）问题或改进：程序中的n的范围有点问题，而且函数的各参数可以改为手动输入，这样可以使程序的可用性更高；（b）小题应该绘制出复指数序列的实部和虚部；同时（a）（b）小题的程序最好分开来方便实验。

（2）修改后代码：

（a）小题：

clc;

clear;

%when:a=-1/12 b=pi/6 K1=1 N=40

%output:the real part and imaginary part of sequence x\_a1[n]

a=input('Type in real part =');

b=input('Type in imaginary part =');

c=a+b\*1i;

K1=input('Type in the gain constant =');

N=input ('Type in length of sequence =');

n=0:N;

x\_a1=K1\*exp(c\*n);

figure,

subplot(121);stem(n,real(x\_a1));%the real part of x\_a1[n]

xlabel('n');ylabel('real(x\_a1)');

title('picture2.23(a):real part of sequence x\_a1[n]');

subplot(122);stem(n,imag(x\_a1));%the imaginary part of x\_a1[n]

xlabel('n');ylabel('imag(x\_a1)');

title('picture2.23(b):imaginary part of sequence x\_a1[n]');

%Figure 2.24(a)

%when:alpha\_a=1.2 K2\_a=0.2

%output:sequence x\_a2\_a[n]

% Figure 2.24(b)

%when:alpha\_b=0.9 K2\_b=20

%output:sequence x\_a2\_b[n]

pause;

N2=input ('Type in length of sequence N2=');

n=0:N2;

alpha\_a=input ('Type in alpha\_a =');

K2\_a=input('Type in K2\_a =');

x\_a2\_a=K2\_a\*alpha\_a.^n;

alpha\_b=input ('Type in alpha\_b =');

K2\_b=input('Type in K2\_b =');

x\_a2\_b=K2\_b\*alpha\_b.^n;

figure,

subplot(121);stem(n,x\_a2\_a);

xlabel('n');ylabel('x\_a2\_a');

title('picture 2.24(a):sequence x\_a2\_a[n]');

subplot(122);stem(n,x\_a2\_b);

xlabel('n');ylabel('x\_a2\_b');

title('picture 2.24(b):sequence x\_a2\_b[n]');

输入：

Type in real part =-1/12

Type in imaginary part =pi/6

Type in the gain constant =1

Type in length of sequence =40

Type in length of sequence N2 =30

Type in alpha\_a =1.2

Type in K2\_a =0.2

Type in alpha\_b =0.9

Type in K2\_b =20

输出：





（b）小题：

%Question(b)----------------------------

%when:K\_b=-2.7 a\_b=-0.4 b\_b=pi/6 N\_b=82

%output:sequence x\_b[n]

clc;

clear;

a\_b=input('Type in real part =');

b\_b=input('Type in imaginary part =');

c\_b=a\_b+b\_b\*1i;

K\_b=input('Type in the gain constant =');

N\_b=input ('Type in length of sequence =');

n\_b=0:N\_b;

x\_b=K\_b\*exp(c\_b\*n\_b);

figure,

subplot(121);stem(n\_b,real(x\_b));%the real part of x\_b[n]

xlabel('n');ylabel('real(x\_b)');

subplot(122);stem(n\_b,imag(x\_b));%the imaginary part of x\_b[n]

xlabel('n');ylabel('imag(x\_b)');

输入：

Type in real part =-0.4

Type in imaginary part =pi/6

Type in the gain constant =-2.7

Type in length of sequence =82

输出：



3. M2.3

（1）问题或改进：将程序改为手动输入函数，这样可以重复使用生成序列。

（2）修改后代码：

clc;

clear;

%when the length of sequence N=100

N=100;

n=0:100;

%Creat the periodic sequence x[n]

x=input('Type in x[n]='); %输入序列

stem(n,x);

xlabel('n');ylabel('x[n]');

title('the periodic sequence x[n]');

（3）运行结果：

输入：Type in x[n]=cos(0.6\*pi\*n+0.3\*pi)

输出：



输入：Type in x[n]=real(exp(pi\*1i\*n/8))+imag(exp(pi\*1i\*n/5))

输出：



输入：Type in x[n]=6\*sin(0.15\*pi\*n)-cos(0.12\*pi\*n+0.1\*pi)

输出：



输入：Type in x[n]=sin(0.1\*pi\*n+0.75\*pi)-3\*cos(0.8\*pi\*n+0.2\*pi)+cos(1.3\*pi\*n)

输出：



4. M2.4

（1）问题或改进：题目要求生成正弦序列x[n]=Asin(w0\*n+phi)而不是x[n]=Acos(w0\*n+phi)。

（2）修改后代码：

A=input('Type in amplitude of sequence=');

L=input('Type in length of sequence=');

omega=input('Type in angular frequency of sequence=');

phi=input('Type in phase of sequence=');

n=0:L;

x=A\*sin(omega\*n+phi);

stem(n,x);

xlabel('n');ylabel('x[n]');

title('sequence x[n]');

（3）运行结果：

图2.22中的序列为x[n]=1.5cos(w0\*n)=1.5sin(w0\*n+pi/2)

例如图（b）

输入：

Type in amplitude of sequence=1.5

Type in length of sequence=40

Type in angular frequency of sequence=0.1\*pi

Type in phase of sequence=pi/2

输出：

5. M2.5

（1）问题或改进：缺少代码注释，且将三个序列画在同一个坐标系中比较好观察比较。

（2）修改后代码：

t=0:0.001:1;

g1=cos(6\*pi\*t);

plot(t,g1,'r'); %画出g1的原图像

hold on;

Ts=0.1;%sampling frequency

t1=0:Ts:1;

g1\_Ts=cos(6\*pi\*t1);

scatter(t1,g1\_Ts,'r'); %画出g1的抽样点

hold on;

g2=cos(14\*pi\*t);

plot(t,g2,'-.g'); %画出g2的原图像

hold on;

g2\_Ts=cos(14\*pi\*t1);

scatter(t1,g2\_Ts,'g'); %画出g2的抽样点

hold on;

g3=cos(26\*pi\*t);

plot(t,g3,'--b'); %画出g2的原图像

hold on;

g3\_Ts=cos(26\*pi\*t1);

scatter(t1,g3\_Ts,'b'); %画出g3的抽样点

xlabel('t');ylabel('Amplitude');

hold off;

（3）运行结果：



6. M2.6

（1）问题后改进：没毛病，不需要修改。

（2）运行结果：

输入：

Type in amplitude of seque1nce=1

Type in phase of sequence=0

Type in length of sequence=1

Type in sampling frequency=0.1

Type in angular frequency vector of sequence=[6\*pi 14\*pi 26\*pi]

输出：



7. M2.7

（1）问题或改进：此题程序可以为一个求互相关序列的程序即可，求自相关序列就是求一个序列自己和自己的互相关序列。

（2）修改后代码：

clear;

clc;

x = input('Type in x1[n]='); %输入第一个序列

a1= input('Type in range\_a1 ='); %输入第一个序列的左边界

b1= input('Type in range\_b1 ='); %输入第一个序列的右边界

y = input('Type in x2[n]='); %输入第二个序列

a2= input('Type in range\_a2 ='); %输入第二个序列的左边界

b2= input('Type in range\_b2 ='); %输入第二个序列的右边界

% Compute the correlation sequence

n1 = a1+a2; n2 = b1+b2;

r = conv(x,fliplr(y));

k = n1:n2;

stem(k,r);xlabel('xy');

（3）运行结果：

输入： y[n]自相关 x[n]和y[n]的互相关

Type in x1[n]=[8 2 -7 -3 0 1 1] Type in x1[n]=[2 0 -1 6 -3 2 0]

Type in range\_a1 =-5 Type in range\_a1 =-3

Type in range\_b1 =1 Type in range\_b1 =3

Type in x2[n]=[8 2 -7 -3 0 1 1] Type in x2[n]=[8 2 -7 -3 0 1 1]

Type in range\_a2 =-5 Type in range\_a2 =-5

Type in range\_b2 =1 Type in range\_b2 =1

输出：



8. M2.8

（1）问题或改进：将图一个一个自上而下排下来显示，导致y轴很短，具体标度很难看清。

（2）修改后代码：

x=input('Type in sequence x[n]=');

x\_index=[-3:3];

y=input('Type in sequence y[n]=');

y\_index=[-5:1];

w=input('Type in sequence w[n]=');

w\_index=[-2:4];

rxx=conv(x,fliplr(x));%autocorrelation sequence

rxx\_index=[min(x\_index)-max(x\_index):max(x\_index)-min(x\_index)];

ryy=conv(x,fliplr(x));%autocorrelation sequence

ryy\_index=[min(y\_index)-max(y\_index):max(y\_index)-min(y\_index)];

rww=conv(x,fliplr(x));%autocorrelation sequence

rww\_index=[min(w\_index)-max(w\_index):max(w\_index)-min(w\_index)];

rxy=conv(x,fliplr(y));%autocorrelation sequence

rxy\_index=[min(x\_index)-max(y\_index):max(x\_index)-min(y\_index)];

rxw=conv(x,fliplr(w));%autocorrelation sequence

rxw\_index=[min(x\_index)-max(w\_index):max(x\_index)-min(w\_index)];

subplot(231);

stem(rxx\_index,rxx);

xlabel('l');ylabel('amplitude');

title('rxx[l]');

subplot(232);

stem(ryy\_index,ryy);

xlabel('l');ylabel('amplitude');

title('ryy[l]');

subplot(233);

stem(rww\_index,rww);

xlabel('l');ylabel('amplitude');

title('rww[l]');

（3）运行结果：

输入：

Type in sequence x[n]=[2 0 -1 6 -3 2 0]

Type in sequence y[n]=[8 2 -7 -3 0 1 1]

Type in sequence w[n]=[3 6 -1 2 6 6 1]

输出：



9. M2.9

（1）问题或改进：没看出问题。

（2）运行结果：

输入：Type in sequence x[n]=[-1 3 2 5 -3 2 6 -4 0 4]

输出：



10. M2.10

（1）问题或改进：可将alpha值和范围n改进为手动输入，增加程序的通用性。

（2）修改后代码：

clc;

clear;

%given attenuation index alpha\_a=0.6 alpha\_b=0.8

%x[n]=(alpha.^n).\*u[n]

%output:two autocorrelation sequence rxxa[l] rxxb[l]

alpha\_a=input('Type in alpha =');

N=input ('Type in length of sequence =');

%when x[n] is a casual sequence

n=0:N;%interception part of the unlimited sequence

xa=alpha\_a.^n;

r=conv(xa,fliplr(xa));

stem(r);

xlabel('l');ylabel('amplitude');

title('r[l]');

（3）运行结果：

输入： Type in alpha =0.6 Type in alpha =0.8

Type in length of sequence =20 Type in length of sequence =20

输出：

 

**Part03 实验**

**题目01**

（1）代码：

z=input('Type in z =');

N=input ('Type in length of sequence =');

n=0:N;

x=exp(z\*n);

subplot(2,1,1);stem(n,real(x));xlabel('n');ylabel('real(x)');

subplot(2,1,2);stem(n,imag(x));xlabel('n');ylabel('imag(x)');

（2）运行结果：

[1] z=1/12+i\*pi/6



[2] z=-1/12+i\*pi/6



[3] z=1/12



[4] z=2+i\*pi/6



[5] z=i\*pi/6



由图像可以看出信号周期为12。

**题目2**

1）代码：

t=0:0.001:1;

f0=input('f0='); %信号频率

A=input('A=');

FT=input('FT='); %抽样频率

g0=A.\*sin(2\*pi\*f0\*t);

subplot(211)

plot(t,g0,'k-'); %绘制原信号

xlabel('t');ylabel('x(t)');

title('x(t)图像');

n=0:1:FT

gs=A.\*sin(2\*pi\*f0\*n/FT);

subplot(212)

stem(n/FT,gs); %绘制抽样信号

xlabel('n');ylabel('x[n]');

title('抽样图像x[n]');

输入：

f0=20 A=1.5 FT=800

输出：



抽样信号频率为20Hz，周期为0.05s。

2）数字频率=2π\*20/fs=0.9，则抽样频率fs=400π/9。没有周期。

输入：

f0=20 A=1.5 FT=400\*pi/9

输出：



**题目3**

1. 单位冲激信号

代码：

clc;

clear;

N=input('N=');

n=-N:N;

h=[(n-0)==0];

stem(n,h);

xlabel('n');ylabel('x[n]');

title('单位冲击响应');

结果：



1. 单位阶跃信号

代码：

clc;

clear;

N=input('N=');

n=-N:N;

h=[(n-0)>=0];

stem(n,h);

xlabel('n');ylabel('x[n]');

title('单位阶跃信号');

结果：



1. 实指数信号

代码：

clc;

clear;

N=input('N=');

A=input('幅度A=');

w=input('底数w=');

n=0:N;

x=A\*w.^n;

stem(n, x)

xlabel('n');ylabel('x[n]');

title('实指数信号');

结果：

输入：N=20 幅度A=0.2 底数w=1.2

输出：



1. 复指数信号

代码：

clc;

clear;

a=input('Type in real part =');

b=input('Type in imaginary part =');

c=a+b\*1i;

A=input('Type in the gain constant =');

N=input ('Type in length of sequence =');

n=0:N;

x =A\*exp(c\*n);

subplot(2,1,1);stem(n,real(x));xlabel('n');ylabel('real(x)');

subplot(2,1,2);stem(n,imag(x));xlabel('n');ylabel('imag(x)');

输入： Type in real part =-1/12

Type in imaginary part =pi/4

Type in the gain constant =1

Type in length of sequence =40

输出：



1. 正弦信号

代码：

clc;

clear;

a=input('Type in w =');

b=input('Type in phi =');

A=input('Type in the gain constant =');

N=input ('Type in length of sequence =');

n=0:N;

x =A\*sin(a\*n+b);

stem(n,real(x));xlabel('n');ylabel('x[n]');title('正弦信号');

输入：

Type in w =pi/10

Type in phi =pi/4

Type in the gain constant =1

Type in length of sequence =50

输出：

