**Name: Hàn Anh Tú**

## Section:

Laboratory Exercise 1

**DISCRETE-TIME SIGNALS: TIME-DOMAIN REPRESENTATION**

## GENERATION OF SEQUENCES

**Project 1.1 Unit sample and unit step sequences**

A copy of Program P1\_1 is given below.

% Program P1\_1

% Generation of a Unit Sample Sequence clf;

% Generate a vector from -10 to 20 n = -10:20;

% Generate the unit sample sequence u = [zeros(1,10) 1 zeros(1,20)];

% Plot the unit sample sequence stem(n,u);

xlabel(TTime index nT);ylabel(TAmplitudeT); title(TUnit Sample SequenceT);

axis([-10 20 0 1.2]);

## Answers:

**Q1.1** The unit sample sequence u[n] generated by running Program P1\_1 is shown below:

Unit Sample Sequence

1

0.8

0.6

Amplitude

0.4

0.2

0

-10 -5 0 5 10 15 20

Time index n

**Q1.2** Công dụng của lệnh clf là – xóa hình vẽ hiện tại.

Công dụng của lệnh axis là – kiểm soát tỷ lệ và hiển thị trục.

Công dụng của lệnh title là – thêm tiêu đề vào biểu đồ hoặc trục và chỉ định thuộc tính văn bản.

Công dụng của lệnh xlabel là – thêm nhãn cho trục x và chỉ định thuộc tính văn bản.

Công dụng của lệnh ylabel là – thêm nhãn cho trục y và chỉ định thuộc tính văn bản

.**Q1.3** The modified Program P1\_1 to generate a delayed unit sample sequence ud[n] with a delay of 11 samples is given below along with the sequence generated by running this program.

% Program P1\_1, MODIFIED for Q1.3

% Generation of a DELAYED Unit Sample Sequence clf;

% Generate a vector from -10 to 20 n = -10:20;

% Generate the DELAYED unit sample sequence u = [zeros(1,21) 1 zeros(1,9)];

% Plot the DELAYED unit sample sequence stem(n,u);

xlabel(TTime index nT);ylabel(TAmplitudeT); title(TDELAYED Unit Sample SequenceT); axis([-10 20 0 1.2]);

DELAYED Unit Sample Sequence

1



0.8

0.6

Amplitude

0.4

0.2

0

-10 -5 0 5 10 15 20

Time index n

**Q1.4** The modified Program P1\_1 to generate a unit step sequence s[n] is given below along with the sequence generated by running this program.

% Program Q1\_4

% Generation of a Unit Step Sequence clf;

% Generate a vector from -10 to 20 n = -10:20;

% Generate the unit step sequence s = [zeros(1,10) ones(1,21)];

% Plot the unit step sequence stem(n,s);

xlabel(TTime index nT);ylabel(TAmplitudeT); title(TUnit Step SequenceT);

axis([-10 20 0 1.2]);

Unit Step Sequence

1

0.8

0.6

Amplitude

0.4

0.2

0

-10 -5 0 5 10 15 20

Time index n

**Q1.5** The modified Program P1\_1 to generate a unit step sequence sd[n] with an advance of 7 samples is given below along with the sequence generated by running this program.

% Program Q1\_5

% Generation of an ADVANCED Unit Step Sequence clf;

% Generate a vector from -10 to 20 n = -10:20;

% Generate the ADVANCED unit step sequence sd = [zeros(1,3) ones(1,28)];

% Plot the ADVANCED unit step sequence stem(n,sd);

xlabel(TTime index nT);ylabel(TAmplitudeT); title(TADVANCED Unit Step SequenceT);

axis([-10 20 0 1.2]);

ADVANCED Unit Step Sequence

1

0.8

0.6

Amplitude

0.4

0.2

0

-10 -5 0 5 10 15 20

Time index n

## Project 1.2 Exponential signals

A copy of Programs P1\_2 and P1\_3 are given below.

% Program P1\_2

% Generation of a complex exponential sequence clf;

c = -(1/12)+(pi/6)\*i; K = 2;

n = 0:40;

x = K\*exp(c\*n); subplot(2,1,1);

stem(n,real(x));

xlabel(TTime index nT);ylabel(TAmplitudeT); title(TReal partT);

subplot(2,1,2);

stem(n,imag(x));

xlabel(TTime index nT);ylabel(TAmplitudeT); title(TImaginary partT);

% Program P1\_3

% Generation of a real exponential sequence clf;

n = 0:35; a = 1.2; K = 0.2;

x = K\*a.^n; stem(n,x);

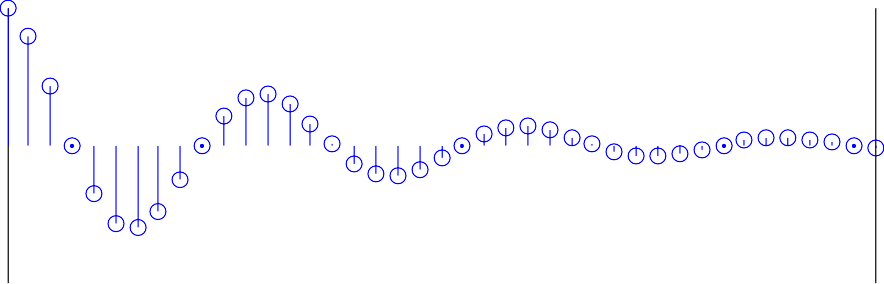
xlabel(TTime index nT);ylabel(TAmplitudeT);

## Answers:

**Q1.6** The complex-valued exponential sequence generated by running Program P1\_2 is shown below:

Real part

2



1

Amplitude

0

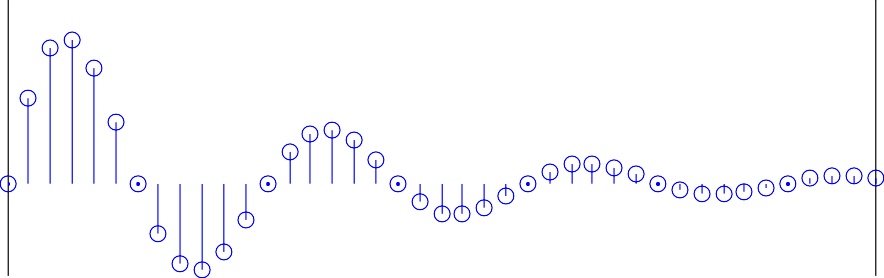
-1

-2

0 5 10 15 20 25 30 35 40

Time index n Imaginary part

2



1

Amplitude

0

-1

0 5 10 15 20 25 30 35 40

Time index n

**Q1.7** Tham số điều khiển tốc độ tăng trưởng hoặc suy giảm của dãy này là - phần thực của c.

Tham số điều khiển biên độ của dãy này là - K.

**Q1.8** Kết quả của việc thay đổi tham số c thành (1/12) + (pi/6)\*i là - vì exp(-1/12) nhỏ hơn một và exp(1/12) lớn hơn một, nên sự thay đổi này đồng nghĩa với việc biên độ của tín hiệu sẽ tăng lên theo n với thay vì suy giảm theo n.

**Q1.9** Mục đích của toán tử real là – để trích xuất phần thực của một vector trong Matlab.

Mục đích của toán tử imag là – để trích xuất phần ảo của một vector trong Matlab..

**Q1.10** Mục đích của lệnh subplot là – để vẽ nhiều đồ thị trong cùng một hình ảnh trong Matlab.

**Q1.11** The real-valued exponential sequence generated by running Program P1\_3 is shown below:

120



100

80

60

Amplitude

40

20

0

0 5 10 15 20 25 30 35

Time index n

**Q1.12** Tham số điều khiển tốc độ tăng trưởng hoặc suy giảm của dãy này là - a.

Tham số điều khiển biên độ của dãy này là - K.

**Q1.13** Sự khác biệt giữa các toán tử số học "^" và ".^" là – "^" dùng để nâng một ma trận vuông lên một lũy thừa bằng phép nhân ma trận. ".^" dùng để nâng mọi phần tử của ma trận hoặc vector lên một lũy thừa; đây là một phép tính "pointwise" hoặc "theo từng điểm" trên từng phần tử.

**Q1.14** The sequence generated by running Program P1\_3 with the parameter a changed to 0.9 and the parameter K changed to 20 is shown below:

20



18

16

14

12

Amplitude

10

8

6

4

2

0

0 5 10 15 20 25 30 35

Time index n

Khi thay đổi tham số a thành một giá trị nhỏ hơn 1, ví dụ như 0.9, và tham số K thành 20, bạn sẽ thu được một dãy số mũ khác với sự suy giảm thay vì sự tăng trưởng.

**Q1.15**

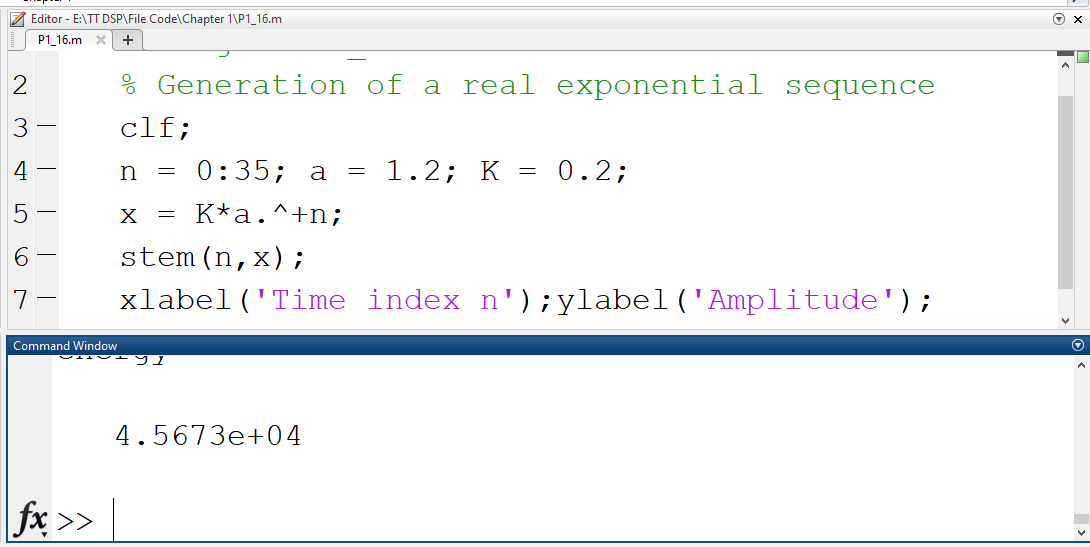
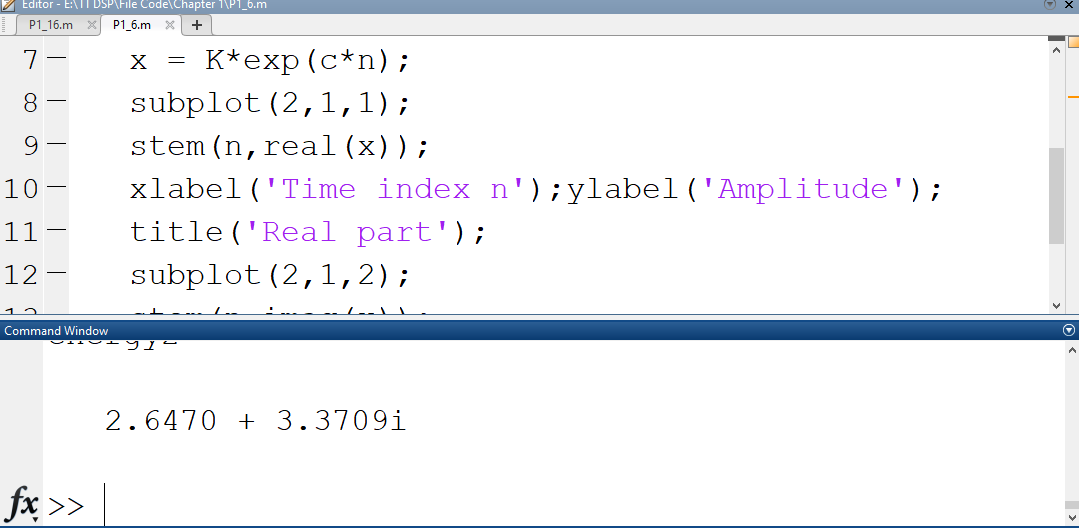
Chiều dài của dãy số này là - 36.

Nó được điều khiển bằng dòng lệnh MATLAB sau đây: n = 0:35;

Nó có thể được thay đổi để tạo ra các dãy số khác nhau với độ dài khác nhau như sau (đưa ra một dòng lệnh ví dụ và chiều dài tương ứng): n = 0:99; tạo ra dãy có chiều dài là 100.

**Q1.16**

Năng lượng của các dãy số mũ có giá trị thực x[n] được tạo ra trong câu hỏi 1.11 và 1.14 và được tính toán bằng lệnh sum lần lượt là - 4.5673e+004 và 2.647e+3.37.

## Project 1.3 Sinusoidal sequences

A copy of Program P1\_4 is given below.

% Program P1\_4

% Generation of a sinusoidal sequence n = 0:40;

f = 0.1;

phase = 0;

A = 1.5;

arg = 2\*pi\*f\*n - phase; x = A\*cos(arg);

clf; % Clear old graph

stem(n,x); % Plot the generated sequence axis([0 40 -2 2]);

grid;

title(TSinusoidal SequenceT); xlabel(TTime index nT); ylabel(TAmplitudeT);

axis;

## Answers:

**Q1.17** The sinusoidal sequence generated by running Program P1\_4 is displayed below.

Sinusoidal Sequence

2



1.5

1

0.5

0

Amplitude

-0.5

-1

-1.5

-2

0 5 10 15 20 25 30 35 40

Time index n

**Q1.18**

Tần số của dãy số này là - f = 0.1 chu kỳ/mẫu.

Nó được điều khiển bằng dòng lệnh MATLAB sau đây: f = 0.1;

Một dãy số có tần số mới là 0.2 có thể được tạo ra bằng dòng lệnh sau:

f = 0.2;

Tham số điều khiển pha của dãy số này là - phase. Tham số điều khiển biên độ của dãy số này là - A. Chu kỳ của dãy số này là - 2π/ω = 1/f = 10.

**Q1.19**

Chiều dài của dãy số này là - 41.

Nó được điều khiển bằng dòng lệnh MATLAB sau đây: n = 0:40;

Một dãy số có chiều dài mới là 81 có thể được tạo ra bằng dòng lệnh sau: n = 0:80;

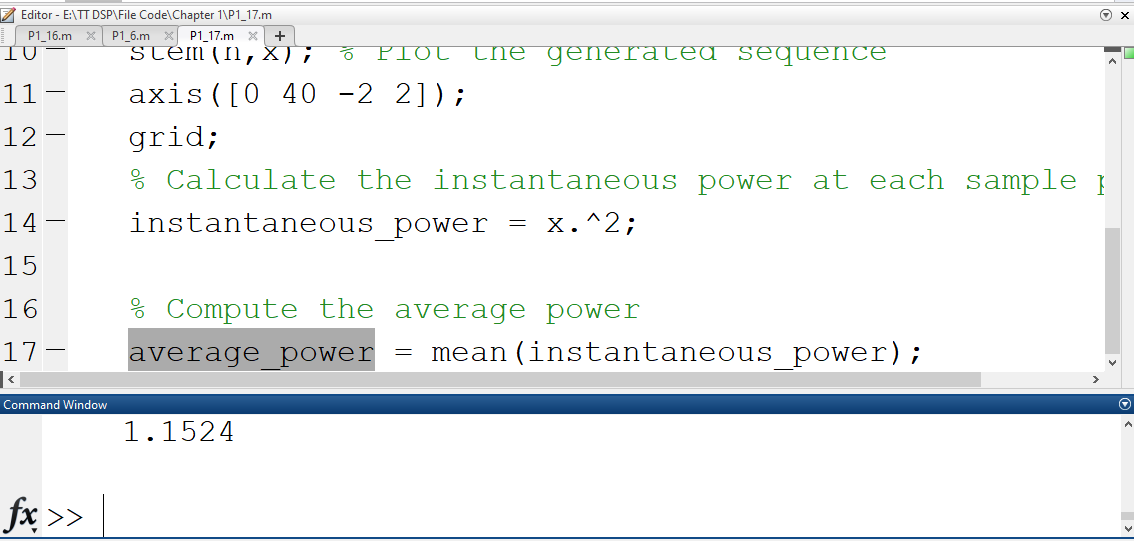
**Q1.20** The average power of the generated sinusoidal sequence is –

% Calculate the instantaneous power at each sample point

instantaneous\_power = x.^2;

% Compute the average power

average\_power = mean(instantaneous\_power);

average\_power = 1.152439024390244

**Q1.21** Mục đích của lệnh axis là – để đặt phạm vi của trục x thành [0,40] và phạm vi của trục y thành [-2,2].

Mục đích của lệnh grid là – để bật việc vẽ các đường lưới (grid lines) trên biểu đồ.

**Q1.22** The modified Program P1\_4 to generate a sinusoidal sequence of frequency 0.9 is given below along with the sequence generated by running it.

% Program Q1\_22A

% Generation of a sinusoidal sequence n = 0:40;

f = 0.9;

phase = 0;

A = 1.5;

arg = 2\*pi\*f\*n - phase; x = A\*cos(arg);

clf; % Clear old graph

stem(n,x); % Plot the generated sequence axis([0 40 -2 2]);

grid;

title(TSinusoidal SequenceT); xlabel(TTime index nT); ylabel(TAmplitudeT);

axis;

Sinusoidal Sequence

2



1.5

1

0.5

0

Amplitude

-0.5

-1

-1.5

-2

0 5 10 15 20 25 30 35 40

Time index n

So sánh giữa dãy số mới với dãy số được tạo ra trong câu hỏi Q1.17 cho thấy - hai biểu đồ là giống nhau. Điều này xảy ra vì trong chương trình được sửa đổi, chúng ta có ω = 0.92π. Điều này tạo ra cùng một đồ thị như một hàm cosine với tần số góc ω - 2π = -0.12π. Vì cosine là một hàm chẵn, điều này tương đương với một hàm cosine có tần số góc +0.1\*2π, giá trị này đã được sử dụng trong P1\_4.m trong câu hỏi Q1.17.

Liên quan đến tần số Hertzian, trong P1\_4.m trong câu hỏi Q1.17, chúng ta có f = 0.1 Hz/mẫu. Trong chương trình được sửa đổi trong câu hỏi Q1.22, chúng ta có f = 0.9 Hz/mẫu, tạo ra cùng một biểu đồ với f = 0.9 - 1 = -0.1. Một lần nữa, vì hàm cosine là chẵn, điều này tạo ra một biểu đồ hoàn toàn giống với biểu đồ chúng ta đã thu được trong Q1.17 với f = +0.1 Hz/mẫu.

A sinusoidal sequence of frequency 1.1 generated by modifying Program P1\_4 is shown below.

Sinusoidal Sequence

2



1.5

1

0.5

0

Amplitude

-0.5

-1

-1.5

-2

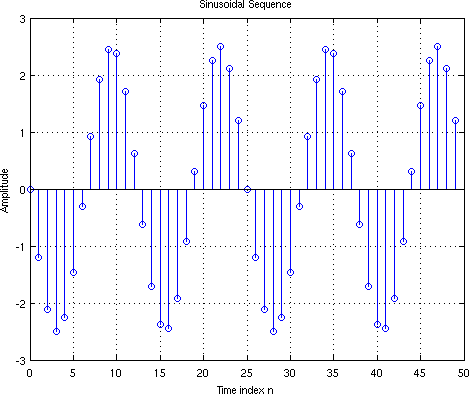
0 5 10 15 20 25 30 35 40

Time index n

So sánh giữa dãy số mới với dãy số được tạo ra trong câu hỏi Q1.17 cho thấy - biểu đồ ở đây lại giống hoàn toàn với biểu đồ trong Q1.17. Điều này xảy ra vì một hàm cosine có tần số f = 1.1 Hz/mẫu giống như một hàm cosine có tần số f = 1.1 - 1 = 0.1 Hz/mẫu, và giá trị này chính là tần số được sử dụng trong Q1.17.

**Q1.23** The sinusoidal sequence of length 50, frequency 0.08, amplitude 2.5, and phase shift of 90 degrees generated by modifying Program P1\_4 is displayed below.

# **NOTE:** Cho chương trình này, cần chuyển đổi góc pha 90 độ sang radian và xem xét dấu trừ xuất hiện trong câu lệnh "arg = 2pifn - phase;" so với dấu cộng được hiển thị trong công thức (1.12) trong tài liệu thực hành. Câu lệnh chính xác để tạo ra góc pha là "phase = -90pi/180;". Cũng cần phải sửa lệnh axis để xem xét chiều dài và biên độ mới của tín hiệu. Câu lệnh axis chính xác là "axis([0 50 -3 3]);".



Chu kỳ của dãy số này là - 2π/ω = 1/f = 1/0.08 = 1/(8/100) = 100/8 = 25/2. Do đó, chu kỳ cơ bản là 25 và biểu đồ có "sự xuất hiện" của việc đi qua 2 chu kỳ của một hình dạng sóng cosine trong mỗi chu kỳ này.

**Q1.24** By replacing the stem command in Program P1\_4 with the plot command, the plot obtained is as shown below:

2

1.5

1

0.5

Amplitude

0

-0.5

-1

-1.5

-2

Sinusoidal Sequence

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

0 5 10 15 20 25 30 35 40

Time index n

Sự khác biệt giữa biểu đồ mới và biểu đồ được tạo ra trong câu hỏi Q1.17 là - thay vì vẽ các ngọn cỏ từ trục x đến các điểm trên đường cong, lệnh "plot" kết nối các điểm bằng các đoạn thẳng, tạo ra sự xấp xỉ của đồ thị của một tín hiệu cosine trong thời gian liên tục.

**Q1.25** By replacing the stem command in Program P1\_4 with the stairs command the plot obtained is as shown below:

Sinusoidal Sequence

2

1.5

1

0.5

0

Amplitude

-0.5

-1

-1.5

-2

0 5 10 15 20 25 30 35 40

Time index n

Sự khác biệt giữa biểu đồ mới và những biểu đồ được tạo ra trong các câu hỏi Q1.17 và Q1.24 là - lệnh "stairs" tạo ra một biểu đồ bước thang (stairstep plot) thay vì biểu đồ ngọn cỏ được tạo ra trong Q1.17 và biểu đồ nối đoạn thẳng được tạo ra trong Q1.24.

## Project 1.4 Random signals Answers:

**Q1.26** The MATLAB program to generate and display a random signal of length 100 with elements

uniformly distributed in the interval [–2, 2] is given below along with the plot of the random sequence generated by running the program:

% Program Q1\_26

n = 0:99;

A = 2;

rng(TshuffleT);

x = 4.\*rand(100,1)-2 ;

clf;

stem(n,x);

axis([0 length(n) -round(2\*(A+0.5))/2 round(2\*(A+0.5))/2]); grid;

title(Tuniform Random SequenceT); xlabel(TTime index nT); ylabel(TAmplitudeT);

axis;

Uniform Random Sequence

2.5



2

1.5

1

0.5

Amplitude

0

-0.5

-1

-1.5

-2

-2.5

0 10 20 30 40 50 60 70 80 90 100

Time index n

**Q1.27** The MATLAB program to generate and display a Gaussian random signal of length 75 with elements normally distributed with zero mean and a variance of 3 is given below along with the plot of the random sequence generated by running the program:

% Program Q1\_27

% Generation of a Gaussian random sequence

% NOTE: if X is a random variable with zero mean and

% unity variance, then (aX + b) is a random variable

% with mean b and variance a^2. This follows from

% basic probability theory. n = 0:74;

xmean = 0; % mean of x

xstd = sqrt(3); % standard deviation of x rng(TshuffleT); % new syntax to seed generator

% generate the sequence

x = xstd\*randn(1,length(n)) + xmean;

% setup the graph and plot

clf; % Clear old graph

stem(n,x); % Plot the generated sequence xmax = max(abs(x));

Ylim = round(2\*(xmax+0.5))/2; axis([0 length(n) -Ylim Ylim]); grid;

title(TGaussian Random SequenceT); xlabel(TTime index nT); ylabel(TAmplitudeT);

axis;

Gaussian Random Sequence

5



4

3

2

1

Amplitude

0

-1

-2

-3

-4

-5

0 10 20 30 40 50 60 70

Time index n

**Q1.28** The MATLAB program to generate and display five sample sequences of a random sinusoidal signal of length 31

# {X[n]} = {A·cos(on + )}

where the amplitude A and the phase  are statistically independent random variables with uniform probability distribution in the range 0  A  4 for the amplitude and in the range 0     for the phase is given below. Also shown are five sample sequences generated by running this program five different times.

% Program Q1\_28

% Generates the "deterministic stochastic process" n = 0:30;

f = 0.1;

Amax = 4; phimax = 2\*pi;

rng(TshuffleT); A = Amax\*rand;

. for trial=1:5

phi = phimax\*rand;

% generate the sequence arg = 2\*pi\*f\*n + phi;

x = A\*cos(arg);

% plot figure(trial);

clf; % Clear any old graph

stem(n,x); % Plot the generated sequence Ylim = round(2\*(Amax+0.5))/2;

axis([0 length(n) -Ylim Ylim]); grid;

title(TSinusoidal Sequence with Random Amplitude and PhaseT); xlabel(TTime index nT);

ylabel(TAmplitudeT); axis;

end

Sinusoidal Sequence with Random Amplitude and Phase

4

3

2

1

0

Amplitude

-1

-2

-3

-4

0 5 10 15 20 25 30

Time index n

Sinusoidal Sequence with Random Amplitude and Phase

4

3

2

1

0

Amplitude

-1

-2

-3

-4

0 5 10 15 20 25 30

Time index n

Sinusoidal Sequence with Random Amplitude and Phase

4

3

2

1

0

Amplitude

-1

-2

-3

-4

0 5 10 15 20 25 30

Time index n

Sinusoidal Sequence with Random Amplitude and Phase

4



3

2

1

0

Amplitude

-1

-2

-3

-4

0 5 10 15 20 25 30

Time index n

Sinusoidal Sequence with Random Amplitude and Phase

4

3

2

1

0

Amplitude

-1

-2

-3

-4

0 5 10 15 20 25 30

Time index n

## SIMPLE OPERATIONS ON SEQUENCES Project 1.5 Signal Smoothing

A copy of Program P1\_5 is given below.

% Program P1\_5

% Signal Smoothing by Averaging clf;

R = 51;

d = 0.8\*(rand(R,1) - 0.5); % Generate random noise m = 0:R-1;

s = 2\*m.\*(0.9.^m); % Generate uncorrupted signal x = s + dT; % Generate noise corrupted signal subplot(2,1,1);

plot(m,dT,Tr-T,m,s,Tg--T,m,x,Tb-.T); xlabel(TTime index nT);ylabel(TAmplitudeT); legend(Td[n] T,Ts[n] T,Tx[n] T);

x1 = [0 0 x];x2 = [0 x 0];x3 = [x 0 0];

y = (x1 + x2 + x3)/3; subplot(2,1,2); plot(m,y(2:R+1),Tr-T,m,s,Tg--T);

legend( Ty[n] T,Ts[n] T);

xlabel(TTime index nT);ylabel(TAmplitudeT);

## Answers:

**Q1.29** The signals generated by running Program P1\_5 are displayed below:

10

d[n]

s[n]

x[n]

5

Amplitude

0

-5

0 5 10 15 20 25 30 35 40 45 50

Time index n

8

y[n]

s[n]

6

Amplitude

4

2

0

0 5 10 15 20 25 30 35 40 45 50

Time index n

**Q1.30** Tín hiệu không bị nhiễu s[n] là - tích của một tăng trưởng tuyến tính với một hàm mũ thực suy giảm chậm.

Nhiễu cộng d[n] là - một dãy số ngẫu nhiên được phân phối đều trong khoảng từ -0.4 đến +0.4..

**Q1.31** Câu lệnh "x = s + d" KHÔNG THỂ được sử dụng để tạo ra tín hiệu bị nhiễu vì - d là một vector cột, trong khi s là một vector hàng; cần phải chuyển vị (transpose) một trong những vector này trước khi cộng chúng.

# **Q1.32** Các mối quan hệ giữa các tín hiệu x1, x2 và x3 và tín hiệu x là như sau:

# Tất cả ba tín hiệu x1, x2 và x3 là phiên bản mở rộng của x, với một mẫu bổ sung được thêm vào phía bên trái và một mẫu bổ sung được thêm vào phía bên phải.

# Tín hiệu x1 là một phiên bản bị trễ của x, bị trễ một mẫu về phía bên phải với việc thêm mẫu giá trị không (zero padding) vào phía bên trái.

# Tín hiệu x2 bằng với x, với việc thêm mẫu giá trị không vào cả phía bên trái và phía bên phải để tạo ra độ dài mở rộng.

# Cuối cùng, x3 là phiên bản tiến thời (shifted) của x, bị trễ một mẫu về phía bên trái với việc thêm mẫu giá trị không vào phía bên phải.

**Q1.33** Mục đích của lệnh legend là – để tạo một hình chú giải (legend) cho các biểu đồ. Trong P1\_5, các tín hiệu được vẽ bằng các màu và kiểu đường khác nhau; hình chú giải cung cấp thông tin về màu và kiểu đường nào được liên kết với mỗi tín hiệu.

## Project 1.6 Generation of Complex Signals

A copy of Program P1\_6 is given below.

% Program P1\_6

% Generation of amplitude modulated sequence clf;

n = 0:100;

m = 0.4;fH = 0.1; fL = 0.01;

xH = sin(2\*pi\*fH\*n); xL = sin(2\*pi\*fL\*n); y = (1+m\*xL).\*xH; stem(n,y);grid;

xlabel(TTime index nT);ylabel(TAmplitudeT);

## Answers:

**Q1.34** T Các tín hiệu biên độ biên độ hóa y[n] được tạo ra bằng cách chạy Chương trình P1\_6 với các giá trị khác nhau của tần số của tín hiệu mang xH[n] và tín hiệu điều chế xL[n], cũng như các giá trị khác nhau của chỉ số điều chế (modulation index) m được thể hiện dưới đây:

# m=0.4; fH=0.1; fL=0.01:

1.5



1

0.5

0

Amplitude

-0.5

-1

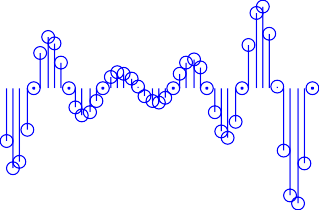
-1.5

0 10 20 30 40 50 60 70 80 90 100

Time index n

# m=0.9; fH=0.1; fL=0.1:

2



1.5

1

0.5

0

Amplitude

-0.5

-1

-1.5

-2

0 10 20 30 40 50 60 70 80 90 100

Time index n

# m=0.4; fH=0.1; fL=0.005:

1.5



1

0.5

0

Amplitude

-0.5

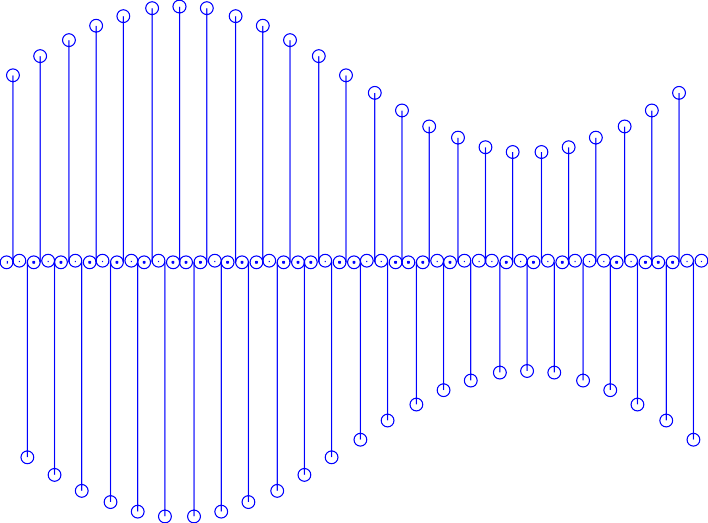
-1

-1.5

0 10 20 30 40 50 60 70 80 90 100

Time index n

# m=0.4; fH=0.25; fL=0.01:

1.5

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

1

0.5

Amplitude

0

-0.5

-1

-1.5

0 10 20 30 40 50 60 70 80 90 100

Time index n

**Q1.35** Sự khác biệt giữa các toán tử số học "" và "." là - "" nhân hai ma trận hoặc vector có thể sử dụng phép nhân ma trận. "." lấy tích điểm của các phần tử của hai ma trận hoặc vector có cùng kích thước.

A copy of Program P1\_7 is given below.

% Program P1\_7

% Generation of a swept frequency sinusoidal sequence n = 0:100;

a = pi/2/100; b = 0;

arg = a\*n.\*n + b\*n; x = cos(arg);

clf; stem(n, x);

axis([0,100,-1.5,1.5]);

title(TSwept-Frequency Sinusoidal SignalT); xlabel(TTime index nT); ylabel(TAmplitudeT);

grid; axis;

## Answers:

**Q1.36** The swept-frequency sinusoidal sequence x[n] generated by running Program P1\_7 is displayed below.

Swept-Frequency Sinusoidal Signal

1.5



1

0.5

0

Amplitude

-0.5

-1

-1.5

0 10 20 30 40 50 60 70 80 90 100

Time index n

**Q1.37** Tần số tối thiểu và tối đa của tín hiệu này là:

Tần số tối thiểu xảy ra tại n=0, nơi chúng ta có 2an+b = 0 rad/mẫu = 0 Hz/mẫu.

Tần số tối đa xảy ra tại n=100, nơi chúng ta có 2an+b = 200a = 200π(0.5)(0.01) = π rad/mẫu = 0.5 Hz/mẫu..

**Q1.38**

Chương trình 1\_7 đã được sửa đổi để tạo ra một tín hiệu sine biên độ dao động với tần số tối thiểu là 0.1 và tần số tối đa là 0.3 được trình bày dưới đây:

Lưu ý: Đối với tần số tối thiểu 0.1 Hz/mẫu = π/5 rad/mẫu tại n=0, chúng ta phải có 2a(0) + b = π/5, điều này ngụ ý rằng b=π/5. Đối với tần số tối đa 0.3 Hz/mẫu = 3π/5 rad/mẫu tại n=100, chúng ta phải có 2a(100) + π/5 = 3π/5, điều này ngụ ý rằng a=π/500.

% Program Q1\_38

% Generation of a swept frequency sinusoidal sequence n = 0:100;

a = pi/500; b = pi/5;

arg = a\*n.\*n + b\*n; x = cos(arg);

clf; stem(n, x);

axis([0,100,-1.5,1.5]);

title(TSwept-Frequency Sinusoidal SignalT); xlabel(TTime index nT); ylabel(TAmplitudeT);

grid; axis;

## WORKSPACE INFORMATION

**Q1.39** Thông tin hiển thị trong cửa sổ lệnh (command window) sau khi thực hiện lệnh who là - một danh sách các tên biến được định nghĩa trong không gian làm việc hiện tại (current workspace).

**Q1.40** Thông tin hiển thị trong cửa sổ lệnh (command window) sau khi thực hiện lệnh whos là - một danh sách chi tiết về các biến được định nghĩa trong không gian làm việc hiện tại, bao gồm tên biến, kích thước của chúng, số byte cần cho mỗi biến và kiểu dữ liệu của từng biến. Ngoài ra, tổng số byte cần cho toàn bộ không gian làm việc cũng được hiển thị.

## OTHER TYPES OF SIGNALS (Optional) Project 1.8 Squarewave and Sawtooth Signals Answer:

**Q1.41** MATLAB programs to generate the square-wave and the sawtooth wave sequences of the type shown in Figures 1.1 and 1.2 are given below along with the sequences generated by running these programs:

% Program Q1\_41A

% Generation of the square wave in Fig. 1.1(a) n = 0:30;

f = 0.1;

phase = 0; duty=60;

A = 2.5;

arg = 2\*pi\*f\*n + phase; x = A\*square(arg,duty);

clf; % Clear old graph

stem(n,x); % Plot the generated sequence axis([0 30 -3 3]);

grid;

title(TSquare Wave Sequence of Fig. 1.1(a)T); xlabel(TTime index nT);

ylabel(TAmplitudeT); axis;

% Program Q1\_41B

% Generation of the square wave in Fig. 1.1(b) n = 0:30;

f = 0.1;

phase = 0; duty=30;

A = 2.5;

arg = 2\*pi\*f\*n + phase; x = A\*square(arg,duty);

clf; % Clear old graph

stem(n,x); % Plot the generated sequence axis([0 30 -3 3]);

grid;

title(TSquare Wave Sequence of Fig. 1.1(b)T); xlabel(TTime index nT);

ylabel(TAmplitudeT); axis;

% Program Q1\_41C

% Generation of the square wave in Fig. 1.2(a) n = 0:50;

f = 0.05;

phase = 0;

peak = 1;

A = 2.0;

arg = 2\*pi\*f\*n + phase;

x = A\*sawtooth(arg,peak);

clf; % Clear old graph

stem(n,x); % Plot the generated sequence axis([0 50 -2 2]);

grid;

title(TSawtooth Wave Sequence of Fig. 1.2(a)T); xlabel(TTime index nT);

ylabel(TAmplitudeT); axis;

% Program Q1\_41D

% Generation of the square wave in Fig. 1.2(b) n = 0:50;

f = 0.05;

phase = 0;

peak = 0.5;

A = 2.0;

arg = 2\*pi\*f\*n + phase;

x = A\*sawtooth(arg,peak);

clf; % Clear old graph

stem(n,x); % Plot the generated sequence axis([0 50 -2 2]);

grid;

title(TSawtooth Wave Sequence of Fig. 1.2(b)T); xlabel(TTime index nT);

ylabel(TAmplitudeT); axis;

Square Wave Sequence of Fig. 1.1(a)

3

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | |  | | | | |  | | | | |  | | | | |  | | | | |  | | | | |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | | | | |  |  |  |  |  |  | | | | |  |  |  |  |  |  | | | | |
|  |  |  |  |  |  | | | | |  |  |  |  |  |  | | | | |  |  |  |  |  |  | | | | |
|  | | | | |  |  |  |  |  |  | | | | |  |  |  |  |  |  | | | | |  |  |  |  |  |
|  | | | | |  |  |  |  |  |  | | | | |  |  |  |  |  |  | | | | |  |  |  |  |  |
|  | | | | |  |  |  |  |  |  | | | | |  |  |  |  |  |  | | | | |  |  |  |  |  |
|  | | | | |  | | | | |  | | | | |



2

1

Amplitude

0

-1

-2



-3

0 5 10 15 20 25 30

Time index n

Square Wave Sequence of Fig. 1.1(b)

3

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | |  | | | | |  | | | | |  | | | | |  | | | | |  | | | | |
|  |  |  | | |  |  |  | | |  |  |  |  | |
|  |  |  | | |  | | | | |  |  |  | | |  | | | | |  |  |  |  | |  | | | | |
|  |  |  | | |  | | | | |  |  |  | | |  | | | | |  |  |  |  | |  | | | | |
|  | | |  |  |  |  |  |  |  |  | | |  |  |  |  |  |  |  |  | | | |  |  |  |  |  |  |
|  | | |  |  |  |  |  |  |  |  | | |  |  |  |  |  |  |  |  | | | |  |  |  |  |  |  |
|  | | |  |  |  |  |  |  |  |  | | |  |  |  |  |  |  |  |  | | | |  |  |  |  |  |  |
|  | | | | |  | | | | |  | | | | |  | | | | |  | | | | |  | | | | |



2

1

Amplitude

0

-1

-2



-3

0 5 10 15 20 25 30

Time index n

Sawtooth Wave Sequence of Fig. 1.2(a)

2



1

0.5

0

Amplitude

-0.5

-1

-1.5

-2

0 5 10 15 20 25 30 35 40 45 50

Time index n

Sawtooth Wave Sequence of Fig. 1.2(b)

2



1.5

1

0.5

0

Amplitude

-0.5

-1

-1.5

-2

0 5 10 15 20 25 30 35 40 45 50

Time index n