



DIGITAL SIGNAL PROCESSING LAB

Lab sheet No. : 01

NAME: NIKHIL KILARI
EE16B017

Roll No:

Question 1

Aim:

To use stem command and plot unit impulse, unit step, ramp and real exponential signals.

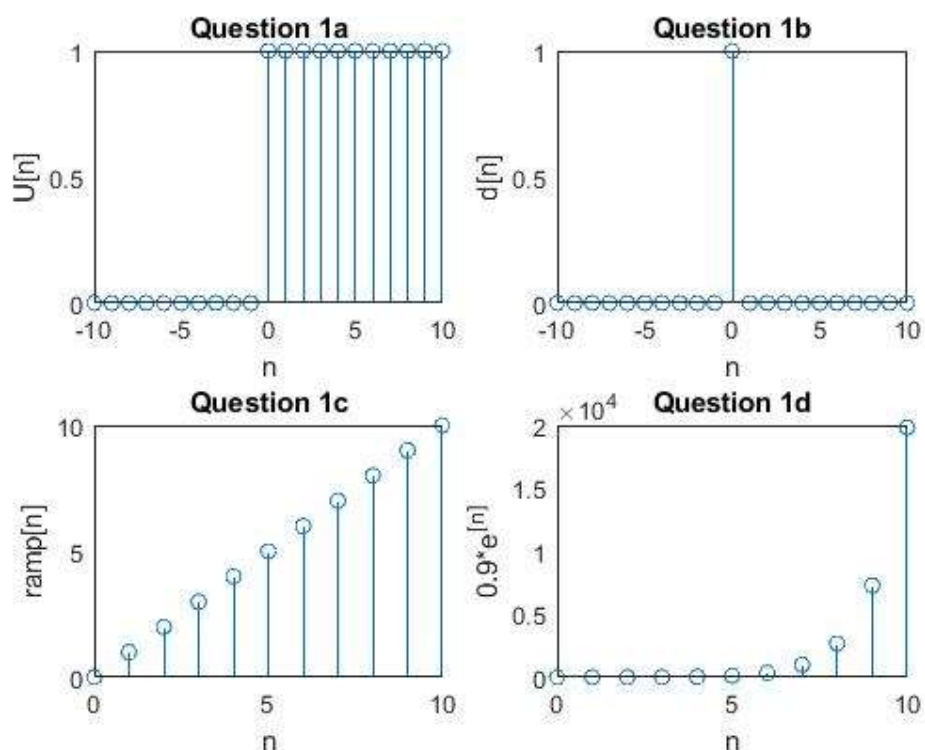
Short Theory:

- a) Unit impulse signal has unit value at origin and zero value at rest of the points.
- b) Unit Step signal has value=1 for $x \geq 0$ and value=0 for $x < 0$.
- c) Ramp signal : $y[n] = n$
- d) Exponential signal : $y[n] = 0.9 * e^x$

Key Commands:

- sympref()
- heaviside()
- subplot
- stem()
- title()
- ylabel(), xlabel()
- exp()

Result:



Inferences/comments:

- `sympref()` – sets the preferred value at the point desired.
- `heaviside()` – For $x > 0$ it returns 1, for $x < 0$ it returns 0, and finally for $x = 0$ it returns $\frac{1}{2}$. To replace $\frac{1}{2}$ with 1, we use `sympref()` to modify the value at origin.
- `exp(x)` returns the exponential values for the array x .

Question 2

Aim:

To generate and plot the following sinusoidal signals by exploring the MATLAB vector handling capability

- a) $x[n] = 3\sin(2\pi n + \pi/3)$ for $-10 \leq n \leq 10$
- b) $x[n] = 5\cos(2\pi n/3 + \pi/4) + 2.5\sin(\pi n/3 + \pi/4)$ for $-10 \leq n \leq 10$

Short Theory:

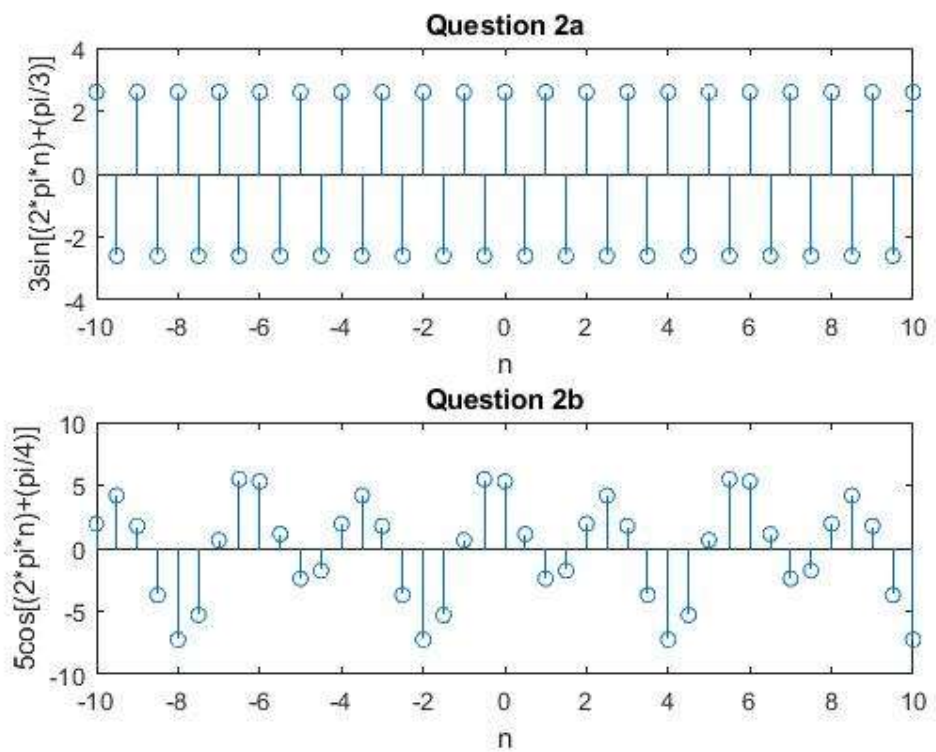
- a) We plot a sin function with amplitude = 3, frequency = 2π , and phase = $\pi/3$
- b) We plot the sum of two sinusoids with amplitude1 = 5, frequency1 = 2π phase1 = π and amplitude2 = 2.5, frequency2 = $\pi/3$, and phase2 = $\pi/4$.

Key Commands:

- subplot
- stem()
- title()
- ylabel(), xlabel()
- sin()

▪ $\cos()$

Result:



Inferences/comments:

- We hereby plot sinusoids, one in which we plot an individual sin function and the other in which we plot the sum of the two sinusoids.

Question 3

Aim:

To find the sum of $x[n] = a^n u[n]$ with $a = 0.9$, where $u[n]$ is the unit step function for the values of x ranging from 0 to 100.

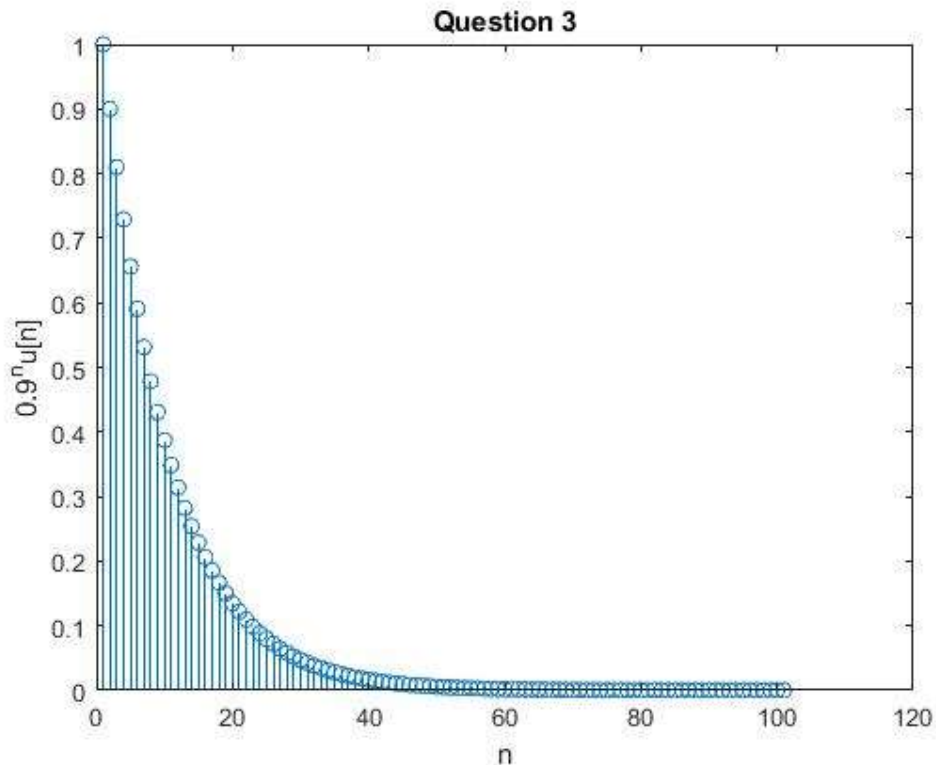
Short Theory:

Here we plot generate the signal $x[n] = a^n u[n]$ with $a = 0.9$ and use the command `sum()` to find the sum of $x[n]$ for values of n ranging from 0 to 100.

Key Commands:

- `subplot`
- `stem()`
- `sympref()`
- `heaviside()`
- `sum()`
- `title()`
- `ylabel(), xlabel()`

Result:



We get the sum of $x[n]$ for n ranging from 0 to 100 as 9.9998.

Analytical Value = 10.

Inferences/comments:

- As mentioned in the previous example
`sympref("HeavisideAtOrigin",1)` sets the value at $n=0$ as 1, thus
 converting the heaviside function into unit step function.

Question 4

Aim:

To plot real, imaginary, absolute and phase of the signal $y[n] = r^n \exp(j\pi n/3)$ with $r = 0.8$ and $r = 1.2$

Short Theory:

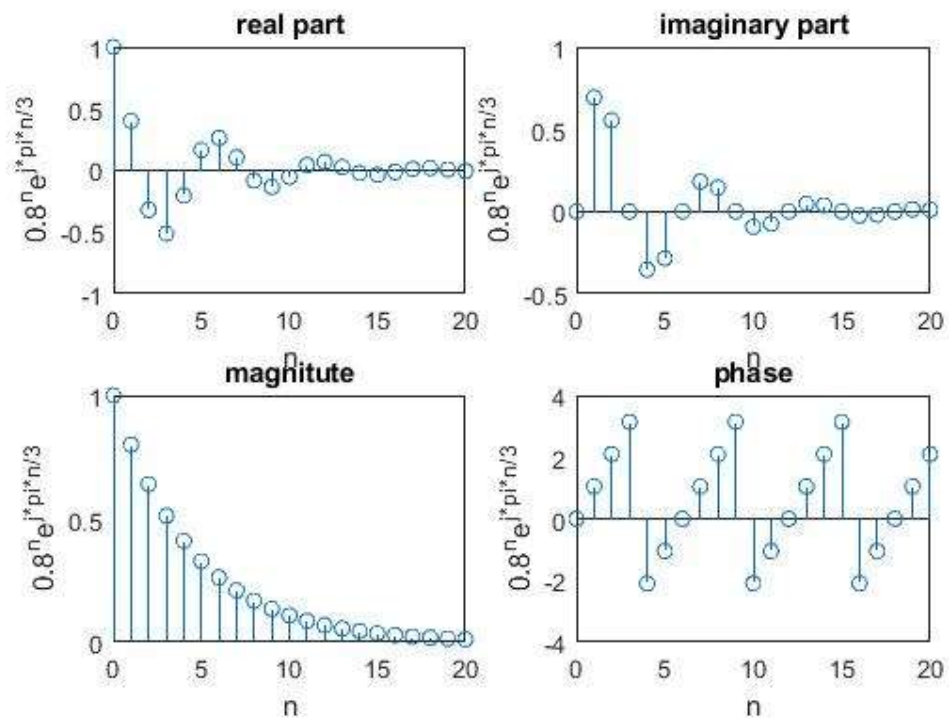
Here we learn how to plot complex signals. There are two ways of plotting the complex signals. One in which we plot real and imaginary parts separately and the second in which plot absolute and phase of the signal separately.

Key Commands:

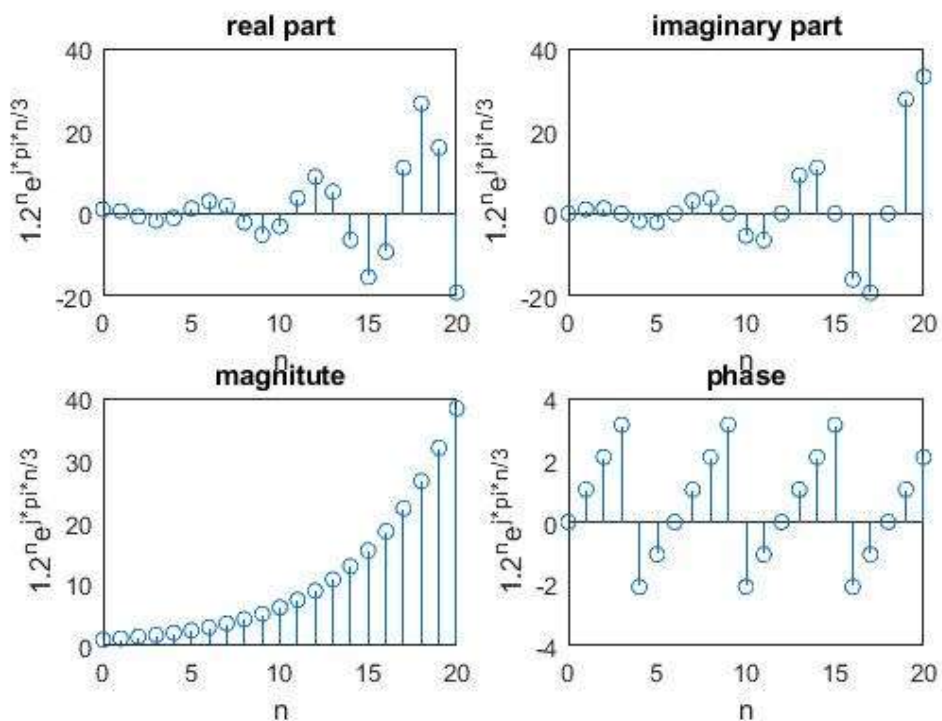
- subplot
- stem()
- exp()
- real()
- imag()
- abs()
- angle()
- title()
- ylabel(), xlabel()

Result:

Question 4 with $r=0.8$



Question 4 with $r=1.2$



Inferences/comments:

- We hereby can verify that the magnitude of the single is exponential and the phase is periodic.

Question 5

Aim:

To plot unit sample sequence and unit step signal and delay unit sample sequence by 11 and advance unit step sequence by 7.

Short Theory:

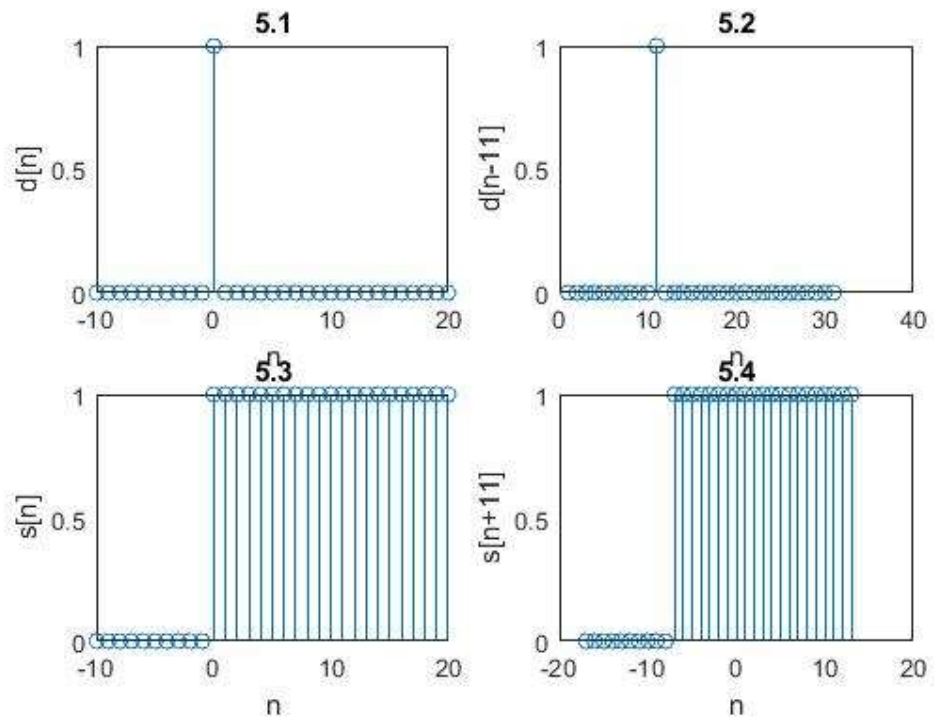
Here we learn how to delay and advance the signals. We plot unit step and sample sequences and delay and advance them respectively as mentioned in the question

Key Commands:

- subplot
- stem(x,y)
- title()
- ylabel(), xlabel()

Result:

Question 5



Inferences/comments:

- If we use `stem(x+11,y)` the signal gets delayed by 11 units, whereas `stem(x-7,y)` the signals gets delayed by 7 units.

Question 6.1

Aim:

To generate a sinusoidal sequence of length 50, frequency 0.08, amplitude 2.5, and phase shift 90 degrees and display it.

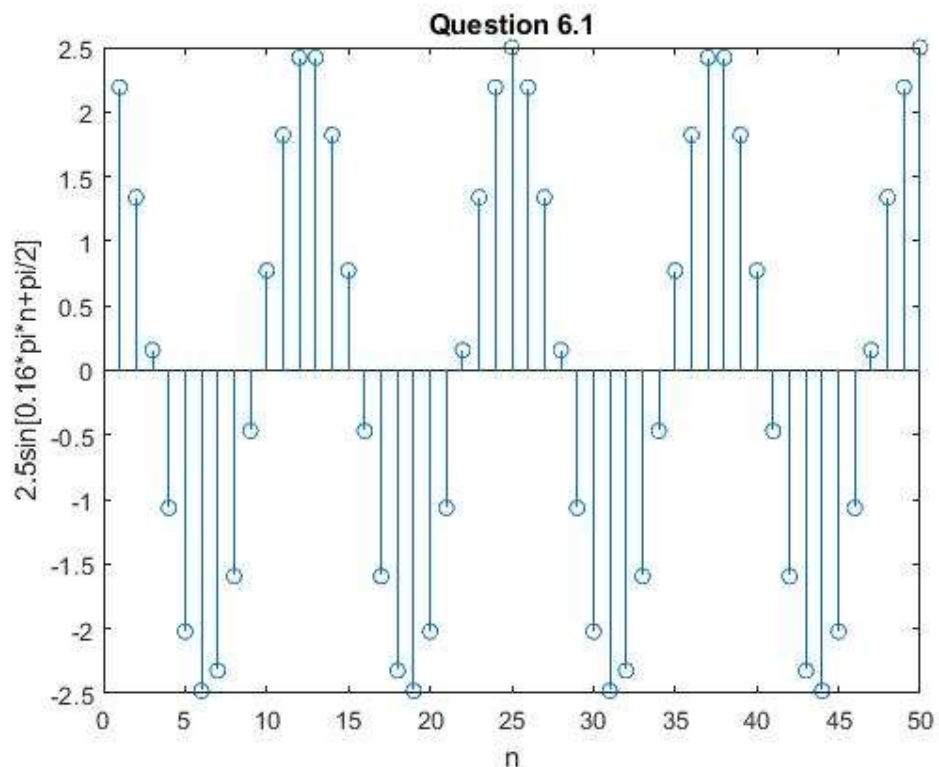
Short Theory:

Here we learn how to plot sinusoidal sequence with the mentioned amplitude, frequency, phase shift and length.

Key Commands:

- subplot
- stem(x,y)
- title()
- ylabel(), xlabel()

Result:



Inferences/comments:

- We hereby plot the specified sinusoidal function as mentioned in the question.

Question 6.2**Aim:**

To create $x_1(t)$ and $x_2(t)$ and to sample $x_3(t) = x_1(t) + x_2(t)$ with a sampling rate 8000Hz in the interval from to 0.1 seconds.

$$x_1(t) = 5 \cos(2\pi * 500t)$$

$$x_2(t) = 5 \cos(2\pi * 1200t + 0.25\pi)$$

Short Theory:

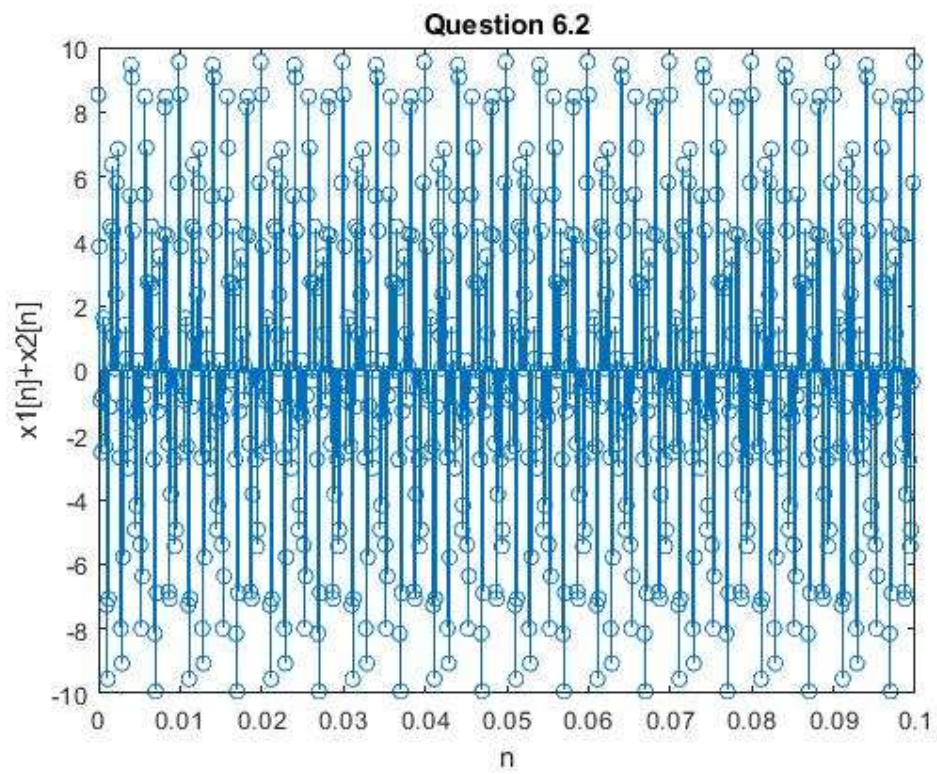
Here we learn how to sample a signal with the specified sampling rate and plot it in the time interval specified in the question

Key Commands:

- subplot
- stem(x,y)
- cos()
- title()

- `ylabel()`, `xlabel()`

Result:



Inferences/comments:

- We hereby successfully sample a continuous time signal with the specified sampling rate and plot it in the range specified in the question.

Question 7.1 and 7.2**Aim:**

7.1: To generate and display a random signal of length 100 whose elements are uniformly distributed in the interval $[-2,2]$.

7.2: To generate and display a Gaussian random signal of length 75 whose elements are normally distributed with zero mean and a variance of 3.

Short Theory:

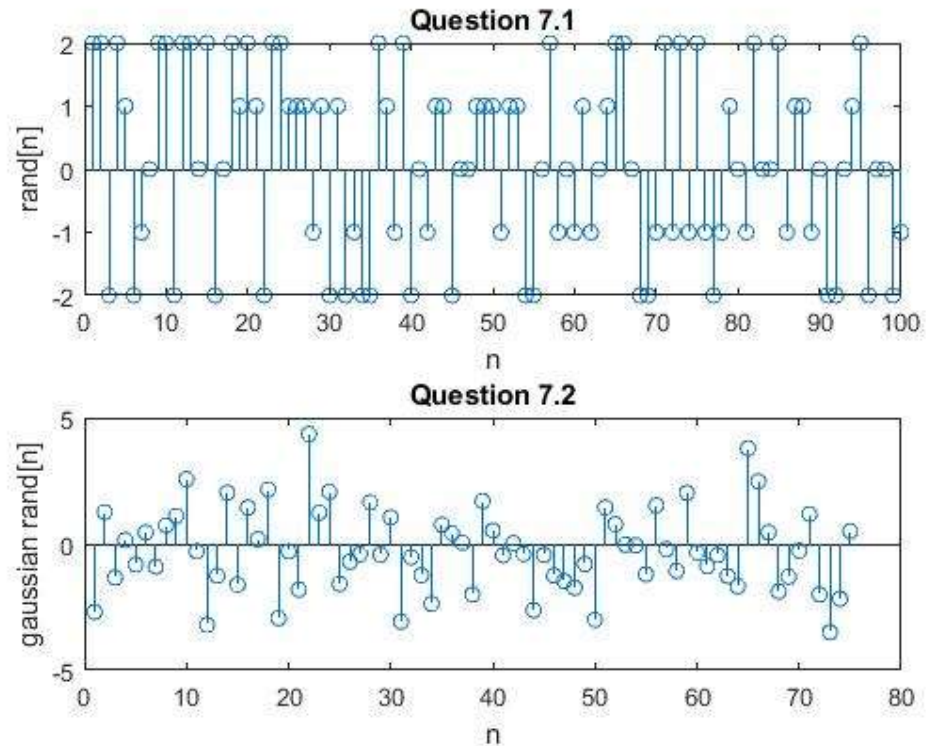
Here we learn how to generate a Gaussian random signal in MATLAB with the required mean and variance.

Key Commands:

- subplot
- stem(x,y)
- randi()
- normrnd()
- title()

- `ylabel()`, `xlabel()`

Result:



Inferences/comments:

- `randi()` generates a uniformly distributed random vector in the range specified.
- `normrnd()` generates a Gaussian random vector with the mentioned mean and variance.

- We therefore successfully plot the two mentioned random vectors with the specifications required as stated in the question.

Question 7.3

Aim:

To generate and display five sample sequences of a random sinusoidal signal of length 31 where A is a statistically independent random variable with uniform probability distribution in the range $[0,4]$ and phase lies in the range $[0,2\pi]$

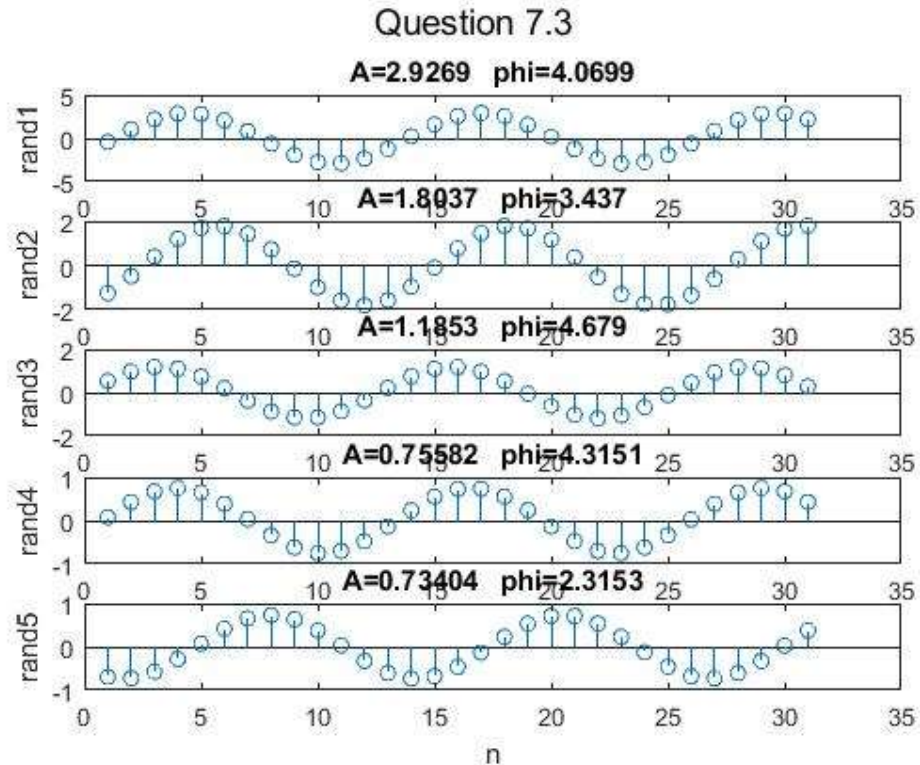
Short Theory:

Here we generate and plot five random signals for the amplitudes and phase in the range specified in the question.

Key Commands:

- subplot
- stem(x,y)
- random()
- cos()
- strcat()
- num2str()
- title()
- ylabel(), xlabel()

Result:



Inferences/comments:

- We hereby plot the five random signals by using a simple for loop
- `strcat()` – concatenates two strings into a single string
- `num2str()` - converts a number into a string
- `random()` – creates the type of random vector as specified in the range required.