**EXPERIMENT 02: Generation of Elementary Signals**

CLASS: BE CMPN A ROLL NO. : 19

Name: Rebecca Dias Pid: 182027

**Aim :** To perform the various elementary signal generation in Python

**Theory :**

**Elementary Discrete Time Signals (Standard DT Signals)**

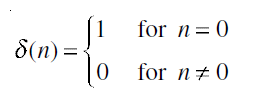
The elementary signals serve as basic building blocks for the construction of more

complex signals. These elementary signals may be used to model a large number of

physical signals, which occur in nature.These elementary signals are also called standard signals.

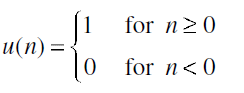
The standard discrete-time signals are as follows:

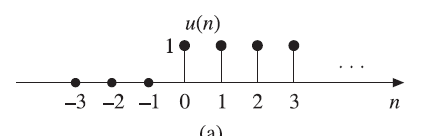
**1. Unit impulse sequence :** The discrete-time unit impulse function 𝝳(n), also called unit sample sequence, is defined as:



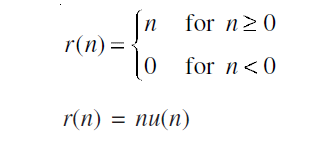
This means that the unit sample sequence is a signal that is zero everywhere, except at n = 0, where its value is unity. It is the most widely used elementary signal for the analysis of signals and systems.

**2.Unit step sequence :** The step sequence is an important signal used for analysis of many discrete-time systems. It exists only for positive time and is zero for negative time.If a step function has unity magnitude, then it is called unit step function. The usefulness of the unit-step function lies in the fact that if we want a sequence to start at n = 0, so that it may have a value of zero for n < 0, we only need to multiply the given sequence with unit step function u(n). The discrete-time unit step sequence u(n) is defined as:





**3. Unit ramp sequence :** The discrete-time unit ramp sequence r(n) is that sequence which starts at n = 0 and increases linearly with time and is defined as:

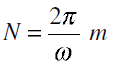


It starts at n = 0 and increases linearly with n.

**4. Sinusoidal Sequence :** The discrete-time sinusoidal sequence is given by



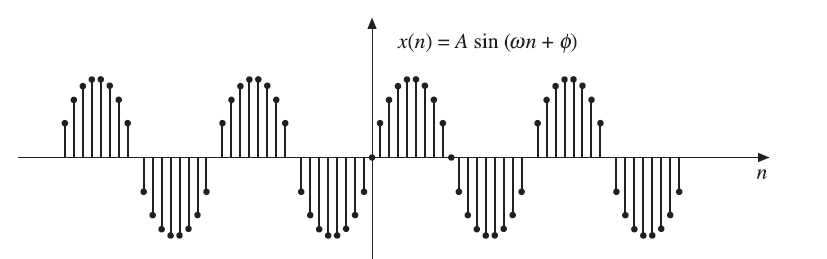
where A is the amplitude, 𝜔 is angular frequency, 𝜙 is phase angle in radians and n is an integer. The period of the discrete-time sinusoidal sequence is:



where N and m are integers.

All continuous-time sinusoidal signals are periodic, but discrete-time sinusoidal

sequences may or may not be periodic depending on the value of 𝜔. For a discrete-time signal to be periodic, the angular frequency 𝜔 must be a rational multiple of 2𝜋. The graphical representation of a discrete-time sinusoidal signal is given as follows:

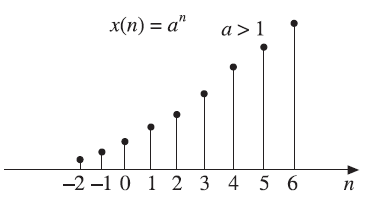


**5. Real Exponential Signals**

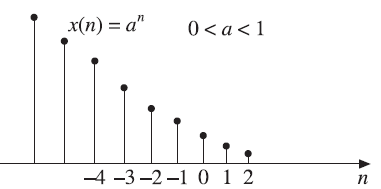
The discrete-time real exponential sequence an is defined as:

x(n) = an for all n

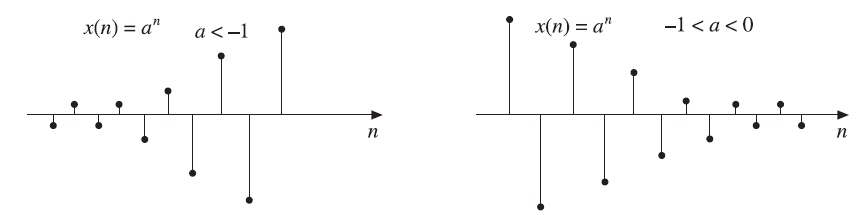
When a > 1, the sequence grows exponentially as shown below



When 0 < a < 1, the sequence decays exponentially as shown below



When a < 0, the sequence takes alternating signs as shown below

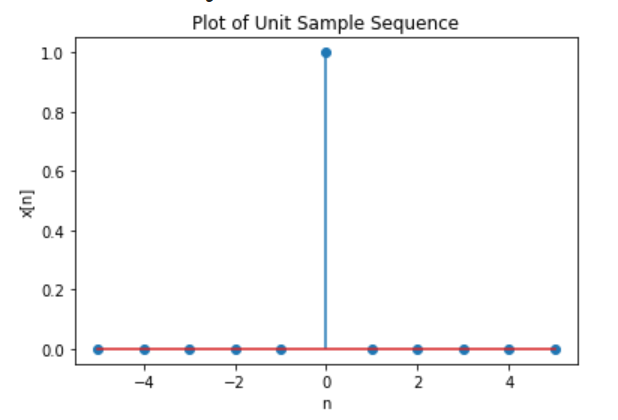


**Programming Exercises in Python**

Unit Impulse Sequence

Code :

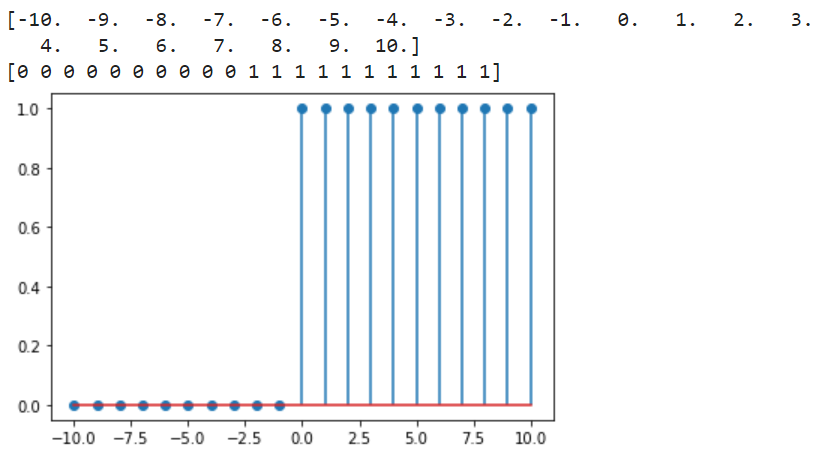
|  |
| --- |
| import numpy as np  import matplotlib.pyplot as plt  n=np.linspace(-5,5,11)  print(n)  x=np.zeros\_like(n)  print(x)  x[5]=1  print(x)  plt.xlabel('n')  plt.ylabel('x[n]')  plt.title('Plot of Unit Sample Sequence')  plt.stem(n,x,use\_line\_collection=True) |



Unit step sequence

Code:

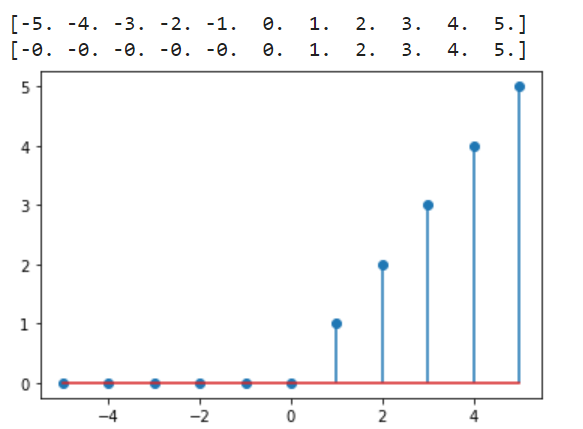
|  |
| --- |
| x=np.linspace(-10,10,21)  y=(x>=0)\*1  print(x)  print(y)  plt.stem(x,y,use\_line\_collection=True)  plt.show() |

****

Unit ramp sequence

Code:

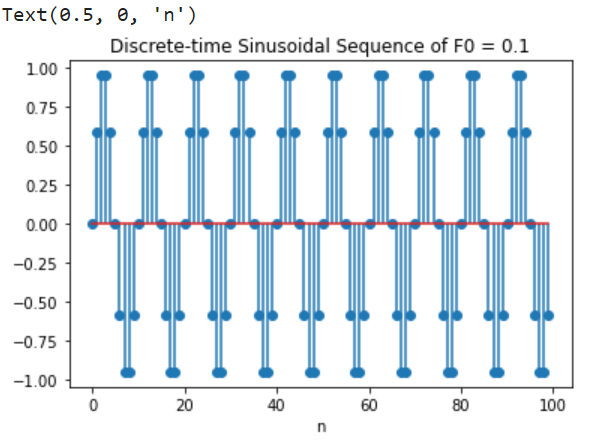
|  |
| --- |
| x=np.linspace(-5,5,11)  y=(x>=0)\*x  print(x)  print(y)  plt.stem(x,y,use\_line\_collection=True)  plt.show() |

****

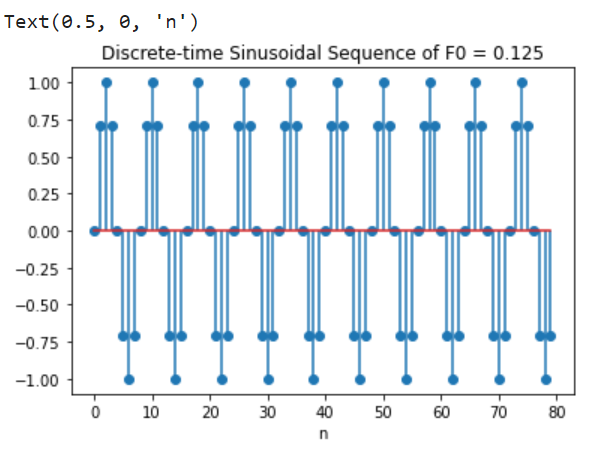
Sinusoidal Sequence

Code:

|  |
| --- |
| F0=0.1  L=100  n=np.arange(L)  x=np.sin(2\*np.pi\*F0\*n)  plt.title('Discrete-time Sinusoidal Sequence of F0 = 0.1')  plt.stem(n,x,use\_line\_collection=True)  plt.xlabel('n') |

****

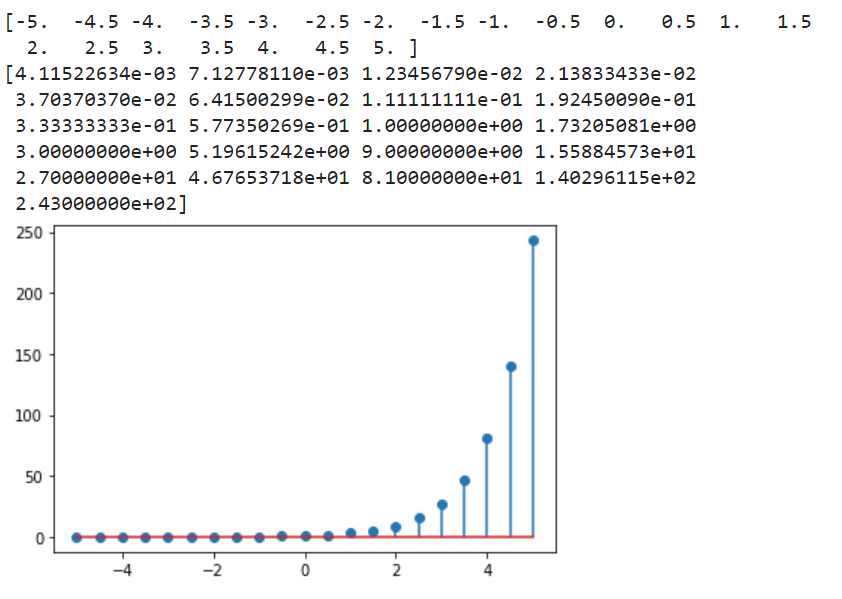
|  |
| --- |
| F0=0.125  L=10\*(1/F0) # Formula is N = 1 / F0  n=np.arange(L)  x=np.sin(2\*np.pi\*F0\*n)  plt.title('Discrete-time Sinusoidal Sequence of F0 = 0.125')  plt.stem(n,x,use\_line\_collection=True)  plt.xlabel('n') |

****

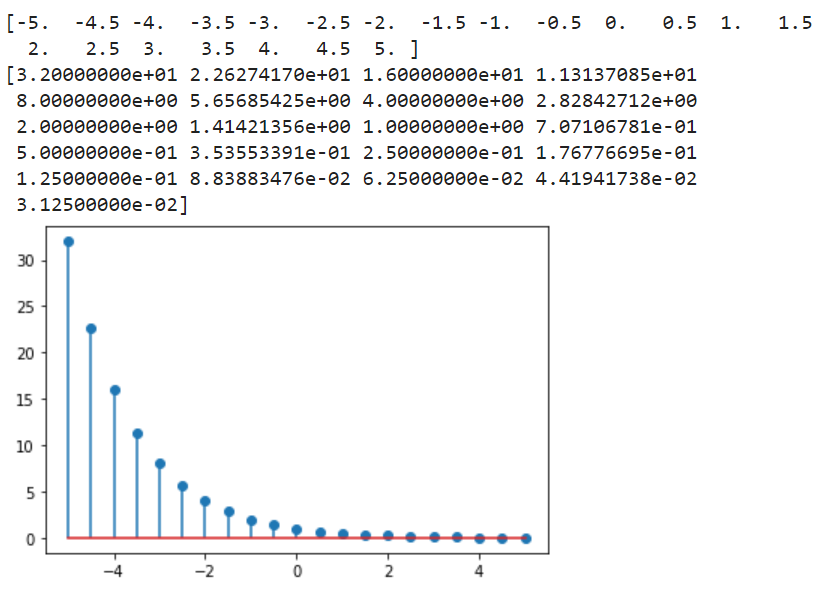
Real Exponential Sequence

Code:

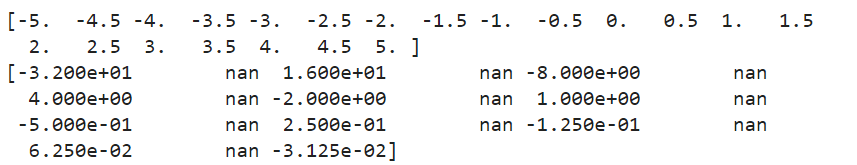
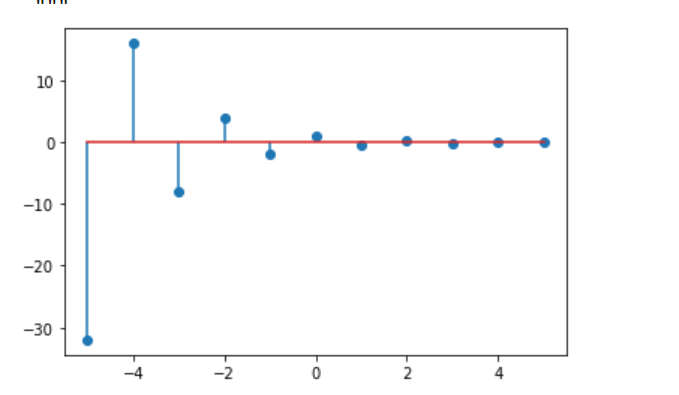
|  |
| --- |
| # a > 1  x=np.linspace(-5,5,21)  # Say a==3  a=3  y=a\*\*x  print(x)  print(y)  plt.stem(x,y,use\_line\_collection=True)  plt.show() |

****

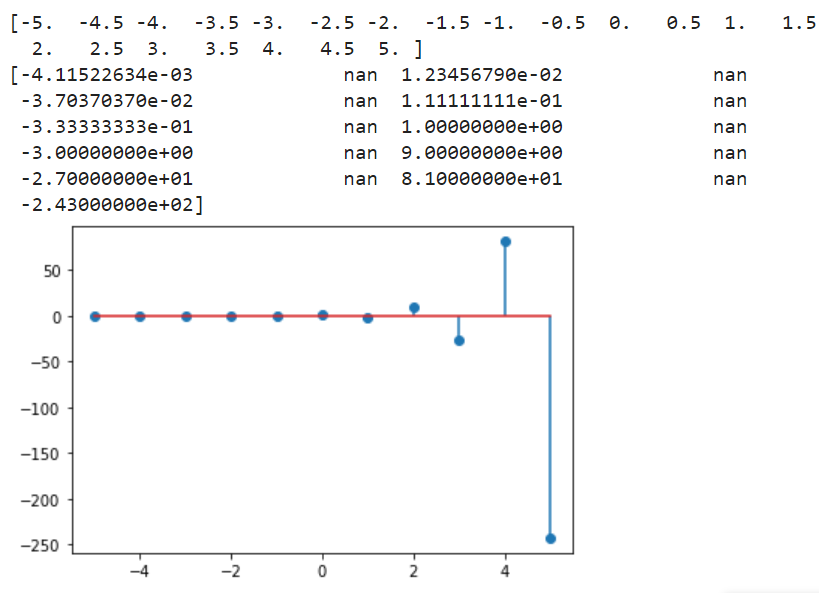
|  |
| --- |
| # 0 < a < 1  x=np.linspace(-5,5,21)  # Say a==0.5  a=0.5  y=a\*\*x  print(x)  print(y)  plt.stem(x,y,use\_line\_collection=True)  plt.show() |

****

|  |
| --- |
| # -1 < a < 0  x=np.linspace(-5,5,21)  #Say a == -0.5  a=-0.5  y=a\*\*x  print(x)  print(y)  plt.stem(x,y,use\_line\_collection=True)  plt.show() |

**** ****

|  |
| --- |
| # a < -1  x=np.linspace(-5,5,21)  # Say a == -3  a=-3  y=a\*\*x  print(x)  print(y)  plt.stem(x,y,use\_line\_collection=True)  plt.show() |

****

**Conclusion:**In this experiment we learnt about the various elementary discrete signals. The various standard discrete signals include- Unit Impulse Signal, Unit Step Signal, Unit Ramp Signal, Sinusoidal and the Exponential signal. We learnt about the functions of each of these signals and plotted their graphs in python as can be seen via the code and output. Thus we successfully completed this experiment.