**DAY 8 ASSIGNMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **26-05-2020** | **Name:** | **Ashish Shanbhag** |
| **Course:** | **DSP** | **USN:** | **4AL16EC008** |
| **Topic:** | **DSP** | **Semester & Section:** | **8th A** |
| **Github Repository:** | **Ashish Shanbhag** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Fourier Transform**  The Fourier transform is a generalization of the [complex](https://mathworld.wolfram.com/ComplexNumber.html) [Fourier series](https://mathworld.wolfram.com/FourierSeries.html) in the limit as L->infty. Replace the discrete A_n with the continuous F(k)dk while letting n/L->k. Then change the sum to an [integral](https://mathworld.wolfram.com/Integral.html), and the equations become    **Fourier Transform and Convolution**  The convolution theorem is a fundamental property of the Fourier transform. It is often stated like Convolution in time domain equals multiplication in frequency domain or Multiplication in time equals convolution in the frequency domain.    The convolution theorem can be beneficially used to compute the convolution of two signals. Rephrasing the convolution theorem, we get    **Laplace Transform of First order**  Laplace transformation is a technique for solving differential equations. Here differential equation of time domain form is first transformed to algebraic equation of frequency domain form. After solving the algebraic equation in frequency domain, the result then is finally transformed to time domain form to achieve the ultimate solution of the differential equation. In other words it can be said that the Laplace transformation is nothing but a shortcut method of solving differential equation**.**    **Application of Z transforms**   * **Sampled Systems :** Inputs and Outputs are related by difference equations and z tansform techniques are used to solve these difference equations. * **Voice transmission:** To band –limit the signal and filter noise from the signal. * Calculation of a signal to control a system.   **Find the Z-Transform of sequence using Matlab** |

**DAY 8 ASSIGNMENT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **26-05-2020** | **Name:** | **Ashish Shanbhag** |
| **Course:** | **PYTHON** | **USN:** | **4AL16EC008** |
| **Topic:** | **Python** | **Semester & Section:** | **8th A** |
| **Github Repository:** | **Ashish Shanbhag** |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS**      **Graphical User Interfaces with Tkinter**  Python offers multiple options for developing GUI (Graphical User Interface). Out of all the GUI methods, tkinter is the most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter is the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.  To create a tkinter app:   * Importing the module – tkinter * Create the main window (container) * Add any number of widgets to the main window * Apply the event Trigger on the widgets.   Importing tkinter is same as importing any other module in the Python code. Note that the name of the module in Python 2.x is ‘Tkinter’ and in Python 3.x it is ‘tkinter’.  **import tkinter**  There are two main methods used which the user needs to remember while creating the Python application with GUI.   * **Tk(screenName=None,  baseName=None,  className=’Tk’,  useTk=1):** To create a main window, tkinter offers a method ‘Tk(screenName=None,  baseName=None,  className=’Tk’,  useTk=1)’. To change the name of the window, you can change the className to the desired one. The basic code used to create the main window of the application is:   m=tkinter.Tk() where m is the name of the main window object   * **mainloop():** There is a method known by the name mainloop() is used when your application is ready to run. mainloop() is an infinite loop used to run the application, wait for an event to occur and process the event as long as the window is not closed.   m.mainloop()  **import tkinter**  **m = tkinter.Tk()**  **'''**  **widgets are added here**  **'''**  **m.mainloop()** |