Assignment 1

G Yashwanth Naik - EE18BTECH11017

Download all python codes from

```
https://github.com/yashwanthguguloth24/EE3025−
    DSP−lab/tree/main/Assignment1/codes
```

and latex-tikz codes from

```
https://github.com/yashwanthguguloth24/EE3025−
    DSP−lab/tree/main/Assignment1
```

1 Problem

1.1. Let

$$x(n) = \left\{\underset{\uparrow}{1}, 2, 3, 4, 2, 1\right\} \tag{1.1.1}$$

$$y(n) + \frac{1}{2}y(n-1) = x(n) + x(n-2) \tag{1.1.2}$$

1.2. Compute

$$X(k) \triangleq \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, \quad k = 0, 1, \dots, N-1 \tag{1.2.1}$$

and $H(k)$ using h(n).

1.3. Compute

$$Y(k) = X(k)H(k) \tag{1.3.1}$$

2 Solution

2.1. We know that,the Impulse Response of the LTI system is the output of the system when Unit Impulse Signal is given as input to the system. So, using Eq (1.1.2) the Impulse Response of the System can be found as,

$$h(n) + \frac{1}{2}h(n-1) = \delta(n) + \delta(n-2) \tag{2.1.1}$$

where $h(n)$ is an IIR Filter.

2.2. DFT of a Input Signal $x(n)$ is

$$X(k) \triangleq \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, \quad k = 0, 1, \dots, N-1 \tag{2.2.1}$$

For N = 6, The above expression can be written in matrix form as below:

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \end{bmatrix} = \begin{bmatrix} W_6^0 W_6^0 W_6^0 & W_6^0 & W_6^0 & W_6^0 \\ W_6^0 W_6^1 W_6^2 & W_6^3 & W_6^4 & W_6^5 \\ W_6^0 W_6^2 W_6^4 & W_6^6 & W_6^8 & W_6^{10} \\ W_6^0 W_6^3 W_6^6 & W_6^9 & W_6^{12} & W_6^{15} \\ W_6^0 W_6^4 W_6^8 & W_6^{12} & W_6^{16} & W_6^{20} \\ W_6^0 W_6^5 W_6^{10} & W_6^{15} & W_6^{20} & W_6^{25} \end{bmatrix} \begin{bmatrix} x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \end{bmatrix} \tag{2.2.2}$$

Where $W_N = e^{-j2\pi/N}$

$$\implies W_6 = e^{-j2\pi/6} = \frac{1 - j\sqrt{3}}{2} \tag{2.2.3}$$

Using $x(n)$ from Eq(1.1.1), we get

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \end{bmatrix} = \begin{bmatrix} W_6^0 W_6^0 W_6^0 & W_6^0 & W_6^0 & W_6^0 \\ W_6^0 W_6^1 W_6^2 & W_6^3 & W_6^4 & W_6^5 \\ W_6^0 W_6^2 W_6^4 & W_6^6 & W_6^8 & W_6^{10} \\ W_6^0 W_6^3 W_6^6 & W_6^9 & W_6^{12} & W_6^{15} \\ W_6^0 W_6^4 W_6^8 & W_6^{12} & W_6^{16} & W_6^{20} \\ W_6^0 W_6^5 W_6^{10} & W_6^{15} & W_6^{20} & W_6^{25} \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 2 \\ 1 \end{bmatrix} \tag{2.2.4}$$

On simplifying we get,

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \end{bmatrix} = \begin{bmatrix} 1+2+3+4+2+1 \\ 1+(2)e^{-j\pi/3}+\ldots+(1)e^{-j5\pi/3} \\ 1+(2)e^{-2j\pi/3}+\ldots+(1)(e^{-2j5\pi/3} \\ 1+(2)e^{-3j\pi/3}+\ldots+(1)e^{-3j5\pi/3} \\ 1+(2)e^{-4j\pi/3}+\ldots+(1)e^{-4j5\pi/3} \\ 1+(2)e^{-5j\pi/3}+\ldots+(1)e^{-5j5\pi/3} \end{bmatrix} \tag{2.2.5}$$

Finally,

$$\begin{bmatrix} X(0)\\ X(1)\\ X(2)\\ X(3)\\ X(4)\\ X(5) \end{bmatrix} = \begin{bmatrix} 13+0j\\ -4-1.732j\\ 1+0j\\ -1+0j\\ 1+0j\\ -4+1.732j \end{bmatrix} \quad (2.2.6)$$

2.3. DFT of a Impulse Response $h(n)$ is

$$H(k) \triangleq \sum_{n=0}^{N-1} h(n)e^{-j2\pi kn/N}, \quad k = 0,1,\dots,N-1 \quad (2.3.1)$$

Similarly, converting the above expression in matrix form to find $H(k)$

$$\begin{bmatrix} H(0)\\ H(1)\\ H(2)\\ H(3)\\ H(4)\\ H(5) \end{bmatrix} = \begin{bmatrix} W_6^0 W_6^0 & W_6^0 & W_6^0 & W_6^0 & W_6^0\\ W_6^0 W_6^1 & W_6^2 & W_6^3 & W_6^4 & W_6^5\\ W_6^0 W_6^2 & W_6^4 & W_6^6 & W_6^8 & W_6^{10}\\ W_6^0 W_6^3 & W_6^6 & W_6^9 & W_6^{12} & W_6^{15}\\ W_6^0 W_6^4 & W_6^8 & W_6^{12} & W_6^{16} & W_6^{20}\\ W_6^0 W_6^5 & W_6^{10} & W_6^{15} & W_6^{20} & W_6^{25} \end{bmatrix} \begin{bmatrix} h(1)\\ h(2)\\ h(3)\\ h(4)\\ h(5)\\ h(6) \end{bmatrix} \quad (2.3.2)$$

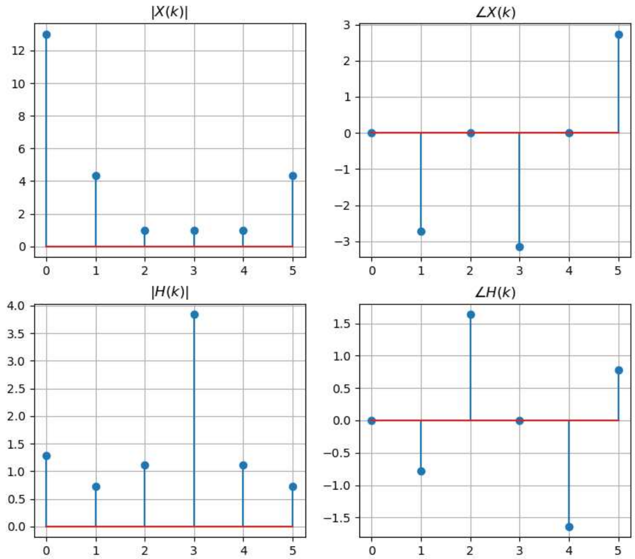
Further Simplification we get,

$$\begin{bmatrix} H(0)\\ H(1)\\ H(2)\\ H(3)\\ H(4)\\ H(5) \end{bmatrix} = \begin{bmatrix} h(0)+h(1)+h(2)+h(3)+h(4)+h(5)\\ h(0)+h(1)e^{-j\pi/3}+\dots+h(5)e^{-j5\pi/3}\\ h(0)+h(1)e^{-2j\pi/3}+\dots+h(5)e^{-2j5\pi/3}\\ h(0)+h(1)e^{-3j\pi/3}+\dots+h(5)e^{-3j5\pi/3}\\ h(0)+h(1)e^{-4j\pi/3}+\dots+h(5)e^{-4j5\pi/3}\\ h(0)+h(1)e^{-5j\pi/3}+\dots+h(5)e^{-5j5\pi/3} \end{bmatrix} \quad (2.3.3)$$

Finally we get,

$$\begin{bmatrix} H(0)\\ H(1)\\ H(2)\\ H(3)\\ H(4)\\ H(5) \end{bmatrix} = \begin{bmatrix} 1.28125+0j\\ 0.51625-0.51418j\\ -0.07812+1.10956j\\ 3.84375+0j\\ -0.07182-1.10956j\\ 0.51625+0.51418j \end{bmatrix} \quad (2.3.4)$$

2.4. The magnitude and phase plots of $X(k)$ and $H(k)$



2.5. We can now compute $Y(k)$ using Eq (2.5.1)

$$Y(k) = X(k)H(k) \quad (2.5.1)$$

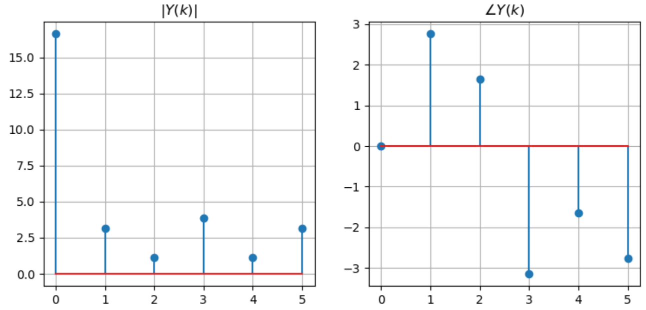
So, $Y(k)$ is obtained element wise multiplication of $X(k)$ and $H(k)$

$$\begin{bmatrix} Y(0)\\ Y(1)\\ Y(2)\\ Y(3)\\ Y(4)\\ Y(5) \end{bmatrix} = \begin{bmatrix} X(0)\cdot H(0)\\ X(1)\cdot H(1)\\ X(2)\cdot H(2)\\ X(3)\cdot H(3)\\ X(4)\cdot H(4)\\ X(5)\cdot H(5) \end{bmatrix} \quad (2.5.2)$$

Computing the above expression we get,

$$\begin{bmatrix} Y(0)\\ Y(1)\\ Y(2)\\ Y(3)\\ Y(4)\\ Y(5) \end{bmatrix} = \begin{bmatrix} 16.6562+0j\\ -2.95312+1.16372j\\ -0.07812+1.10959j\\ -3.84375-9.27556j\\ -0.07812-1.10959j\\ -2.95312-1.16372j \end{bmatrix} \quad (2.5.3)$$

The magnitude and phase plots of $Y(k)$ are



2.6. The following code plots all the above figures.

https://github.com/yashwanthguguloth24/
EE3025−DSP−lab/tree/main/
Assignment1/codes/ee18btech11017_1.py

3 DFT Properties

3.1. *Properties*

a) Symmetric Property :

$$W_N^{k+N/2} = -W_N^k$$

b) Periodic Property :

$$W_N^{k+N} = W_N^k$$

c) Square Property :

$$W_N^2 = W_{N/2}$$

3.2. For a N-point DFT we can write,

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk}, \quad k = 0, 1, \ldots, N-1 \tag{3.2.1}$$

Dividing the inputs into even and odd indices and using Property(c),

$$X(k) = \sum_{n=even} x(n)W_N^{kn} + \sum_{n=odd} x(n)W_N^{kn} \tag{3.2.2}$$

$$= \sum_{m=0}^{N/2-1} x(2m)W_N^{2mk} + \sum_{m=0}^{N/2-1} x(2m+1)W_N^{(2m+1)k} \tag{3.2.3}$$

$$= \sum_{m=0}^{N/2-1} x(2m)W_{N/2}^{mk} + W_N^k \sum_{m=0}^{N/2-1} x(2m+1)W_{N/2}^{mk} \tag{3.2.4}$$

Finally,

$$X(k) = \underbrace{X_e(k)}_{\substack{\text{N/2 point DFT}\\ \text{with even inputs}}} + W_N^k \underbrace{X_o(k)}_{\substack{\text{N/2 point DFT}\\ \text{with odd inputs}}} \tag{3.2.5}$$

Taking N = 6 and expressing the even odd DFT's $X_e(k)$, $X_o(k)$ interms of matrices we get,

$$\begin{bmatrix} X_e(0) \\ X_e(1) \\ X_e(2) \\ X_o(0) \\ X_o(1) \\ X_o(2) \end{bmatrix} = \begin{bmatrix} W_3^0 & W_3^0 & W_3^0 & 0 & 0 & 0 \\ W_3^0 & W_3^1 & W_3^2 & 0 & 0 & 0 \\ W_3^0 & W_3^2 & W_3^4 & 0 & 0 & 0 \\ 0 & 0 & 0 & W_3^0 & W_3^0 & W_3^0 \\ 0 & 0 & 0 & W_3^0 & W_3^1 & W_3^2 \\ 0 & 0 & 0 & W_3^0 & W_3^2 & W_3^4 \end{bmatrix} \begin{bmatrix} x(0) \\ x(2) \\ x(4) \\ x(1) \\ x(3) \\ x(5) \end{bmatrix} \tag{3.2.6}$$

Let, F_N be a N-Point DFT Matrix and P_N is an odd-even Permutation Matrix, we can write above equation as

$$\begin{bmatrix} X_e(0) \\ X_e(1) \\ X_e(2) \\ X_o(0) \\ X_o(1) \\ X_o(2) \end{bmatrix} = \begin{bmatrix} F_3 & 0 \\ 0 & F_3 \end{bmatrix} P_6 x \tag{3.2.7}$$

where

$$P_6 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.2.8}$$

and

$$P_6 x = P_6 \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \end{bmatrix} = \begin{bmatrix} x(0) \\ x(2) \\ x(4) \\ x(1) \\ x(3) \\ x(5) \end{bmatrix} \tag{3.2.9}$$

Now,using (3.2.5) we can express $X(k)$ interms of $X_e(k)$ and $X_o(k)$

$$\begin{bmatrix}X(0)\\X(1)\\X(2)\\X(3)\\X(4)\\X(5)\end{bmatrix} = \begin{bmatrix}1&0&0&W_6^0&0&0\\0&1&0&0&W_6^1&0\\0&0&1&0&0&W_6^2\\1&0&0&W_6^3&0&0\\0&1&0&0&W_6^4&0\\0&0&1&0&0&W_6^5\end{bmatrix}\begin{bmatrix}X_e(0)\\X_e(1)\\X_e(2)\\X_o(0)\\X_o(1)\\X_o(2)\end{bmatrix} \tag{3.2.10}$$

Considering I_3 to be 3x3 identity matrix and $D_N = diag(1, W_N, W_N^2, ..., W_N^{N-1})$. So that $D_{\frac{N}{2}} = diag(1, W_N, W_N^2, ..., W_N^{\frac{N}{2}-1})$. Therefore D_3 will be,

$$D_3 = \begin{bmatrix}1&0&0\\0&W_6^1&0\\0&0&W_6^2\end{bmatrix} \tag{3.2.11}$$

Using Property(a), Equation (3.2.10) and above defined D_3 and I_3 we can write,

$$X = \begin{bmatrix}X(0)\\X(1)\\X(2)\\X(3)\\X(4)\\X(5)\end{bmatrix} = \begin{bmatrix}I_3 & D_3\\I_3 & -D_3\end{bmatrix}\begin{bmatrix}X_e(0)\\X_e(1)\\X_e(2)\\X_o(0)\\X_o(1)\\X_o(2)\end{bmatrix} \tag{3.2.12}$$

Finally using Eq (3.2.12) and Eq (3.2.7) we get

$$X = \begin{bmatrix}I_3 & D_3\\I_3 & -D_3\end{bmatrix}\begin{bmatrix}F_3 & 0\\0 & F_3\end{bmatrix}P_6x \tag{3.2.13}$$

Using $X = F_6x$ for N = 6;

$$\implies F_6 = \begin{bmatrix}I_3 & D_3\\I_3 & -D_3\end{bmatrix}\begin{bmatrix}F_3 & 0\\0 & F_3\end{bmatrix}P_6 \tag{3.2.14}$$

Therefore, for an arbitary N we can express N-point DFT Matrix in terms of N/2-point DFT Matrix as

$$F_N = \begin{bmatrix}I_{N/2} & D_{N/2}\\I_{N/2} & -D_{N/2}\end{bmatrix}\begin{bmatrix}F_{N/2} & 0\\0 & F_{N/2}\end{bmatrix}P_N \tag{3.2.15}$$

3.3. Now, if $N = 2^M$ where $M \in \mathbb{Z}^+$ then we can recursively breakdown N/2 point DFT Matrix to N/4 point DFT Matrix ..so on till we reach 2-point DFT Matrix.So for N = 8, using Eq (3.2.15) we can write,

$$F_8 = \begin{bmatrix}I_4 & D_4\\I_4 & -D_4\end{bmatrix}\begin{bmatrix}F_4 & 0\\0 & F_4\end{bmatrix}P_8 \tag{3.3.1}$$

$$F_4 = \begin{bmatrix}I_2 & D_2\\I_2 & -D_2\end{bmatrix}\begin{bmatrix}F_2 & 0\\0 & F_2\end{bmatrix}P_4 \tag{3.3.2}$$

Finally, the 2-point DFT Matrix is the base case

$$F_2\begin{bmatrix}x_1\\x_2\end{bmatrix} = \begin{bmatrix}1&1\\1&-1\end{bmatrix}\begin{bmatrix}x_1\\x_2\end{bmatrix} = \begin{bmatrix}x_1+x_2\\x_1-x_2\end{bmatrix} \tag{3.3.3}$$

3.4. Step by Step visualization of computing 8-Point DFT recursively using 4-point DFT's and 2-point DFT's.Expressing 8-point DFT's in terms of 4-point DFT's.

$$\begin{bmatrix}X(0)\\X(1)\\X(2)\\X(3)\end{bmatrix} = \begin{bmatrix}X_e(0)\\X_e(1)\\X_e(2)\\X_e(3)\end{bmatrix} + \begin{bmatrix}W_8^0&0&0&0\\0&W_8^1&0&0\\0&0&W_8^2&0\\0&0&0&W_8^3\end{bmatrix}\begin{bmatrix}X_o(0)\\X_o(1)\\X_o(2)\\X_o(3)\end{bmatrix} \tag{3.4.1}$$

$$\begin{bmatrix}X(4)\\X(5)\\X(6)\\X(7)\end{bmatrix} = \begin{bmatrix}X_e(0)\\X_e(1)\\X_e(2)\\X_e(3)\end{bmatrix} - \begin{bmatrix}W_8^0&0&0&0\\0&W_8^1&0&0\\0&0&W_8^2&0\\0&0&0&W_8^3\end{bmatrix}\begin{bmatrix}X_o(0)\\X_o(1)\\X_o(2)\\X_o(3)\end{bmatrix} \tag{3.4.2}$$

Now, 4-point DFT's to 2-point DFT's

$$\begin{bmatrix}X_e(0)\\X_e(1)\end{bmatrix} = \begin{bmatrix}X_{e_1}(0)\\X_{e_1}(1)\end{bmatrix} + \begin{bmatrix}W_4^0&0\\0&W_4^1\end{bmatrix}\begin{bmatrix}X_{o_1}(0)\\X_{o_1}(1)\end{bmatrix} \tag{3.4.3}$$

$$\begin{bmatrix}X_e(2)\\X_e(3)\end{bmatrix} = \begin{bmatrix}X_{e_1}(0)\\X_{e_1}(1)\end{bmatrix} - \begin{bmatrix}W_4^0&0\\0&W_4^1\end{bmatrix}\begin{bmatrix}X_{o_1}(0)\\X_{o_1}(1)\end{bmatrix} \tag{3.4.4}$$

$$\begin{bmatrix}X_o(0)\\X_o(1)\end{bmatrix} = \begin{bmatrix}X_{e_2}(0)\\X_{e_2}(1)\end{bmatrix} + \begin{bmatrix}W_4^0&0\\0&W_4^1\end{bmatrix}\begin{bmatrix}X_{o_2}(0)\\X_{o_2}(1)\end{bmatrix} \tag{3.4.5}$$

$$\begin{bmatrix}X_o(2)\\X_o(3)\end{bmatrix} = \begin{bmatrix}X_{e_2}(0)\\X_{e_2}(1)\end{bmatrix} - \begin{bmatrix}W_4^0&0\\0&W_4^1\end{bmatrix}\begin{bmatrix}X_{o_2}(0)\\X_{o_2}(1)\end{bmatrix} \tag{3.4.6}$$

$$P_8\begin{bmatrix}x(0)\\x(1)\\x(2)\\x(3)\\x(4)\\x(5)\\x(6)\\x(7)\end{bmatrix} = \begin{bmatrix}x(0)\\x(2)\\x(4)\\x(6)\\x(1)\\x(3)\\x(5)\\x(7)\end{bmatrix} \tag{3.4.7}$$

$$P_4\begin{bmatrix}x(0)\\x(2)\\x(4)\\x(6)\end{bmatrix} = \begin{bmatrix}x(0)\\x(4)\\x(2)\\x(6)\end{bmatrix} \tag{3.4.8}$$

$$P_4 \begin{bmatrix} x(1) \\ x(3) \\ x(5) \\ x(7) \end{bmatrix} = \begin{bmatrix} x(1) \\ x(5) \\ x(3) \\ x(7) \end{bmatrix} \tag{3.4.9}$$

Finally,

$$\begin{bmatrix} X_{e_1}(0) \\ X_{e_1}(1) \end{bmatrix} = F_2 \begin{bmatrix} x(0) \\ x(4) \end{bmatrix} = \begin{bmatrix} x(0)+x(4) \\ x(0)-x(4) \end{bmatrix} \tag{3.4.10}$$

$$\begin{bmatrix} X_{o_1}(0) \\ X_{o_1}(1) \end{bmatrix} = F_2 \begin{bmatrix} x(2) \\ x(6) \end{bmatrix} = \begin{bmatrix} x(2)+x(6) \\ x(2)-x(6) \end{bmatrix} \tag{3.4.11}$$

$$\begin{bmatrix} X_{e_2}(0) \\ X_{e_2}(1) \end{bmatrix} = F_2 \begin{bmatrix} x(1) \\ x(5) \end{bmatrix} = \begin{bmatrix} x(1)+x(5) \\ x(1)-x(5) \end{bmatrix} \tag{3.4.12}$$

$$\begin{bmatrix} X_{o_2}(0) \\ X_{o_2}(1) \end{bmatrix} = F_2 \begin{bmatrix} x(3) \\ x(7) \end{bmatrix} = \begin{bmatrix} x(3)+x(7) \\ x(3)-x(7) \end{bmatrix} \tag{3.4.13}$$

So, $X_{e_2} \in \text{DFT}\{x(1),x(5)\}$ and $X_{o_2} \in \text{DFT}\{x(3),x(7)\}$ would combine to give X_o .And $X_{e_1} \in \text{DFT}\{x(0),x(4)\}$ and $X_{o_1} \in \text{DFT}\{x(2),x(6)\}$ would combine to give X_e.

3.5. *Time Complexity:*

In DFT - Matrix multiplication of NxN matrix with Nx1 vector. Hence it has $O(N^2)$ time complexity which is very slow for high N.

In this recursive approach which is termed as FFT - N-point FFT is broken down recursively into 2 N/2-point FFTs recursively. Additionally $O(N)$ operation of Vector multiplication is performed on the N/2 point FFTs.

$$T(n) = 2T(n/2) + O(n) \tag{3.5.1}$$

Solving this recurrence relation gives $O(NlogN)$ time complexity.

3.6. Computing $X(k)$, $H(k)$ and $Y(k)$ for

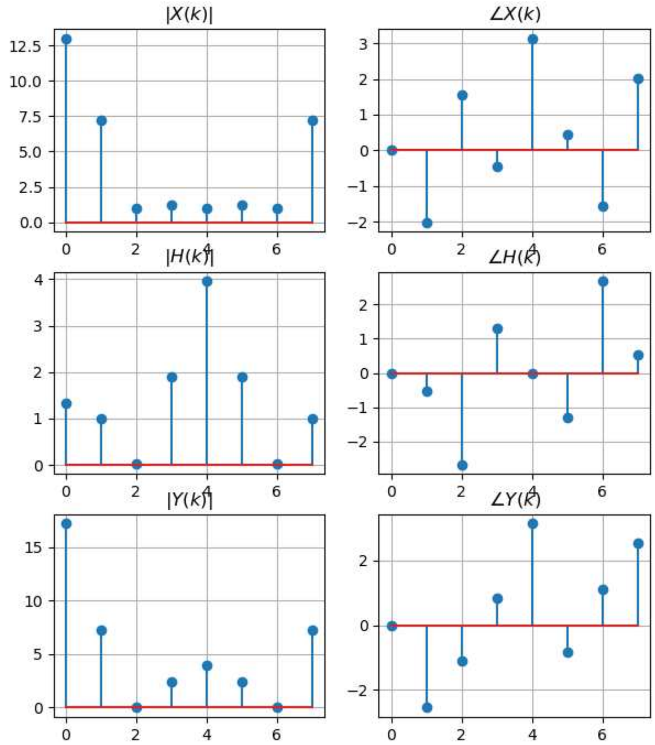
$$x(n) = \left\{ \underset{\uparrow}{1}, 2, 3, 4, 2, 1, 0, 0 \right\} \tag{3.6.1}$$

with N = 8, using above FFT approach.

We can observe that both the approaches produces that same plots.

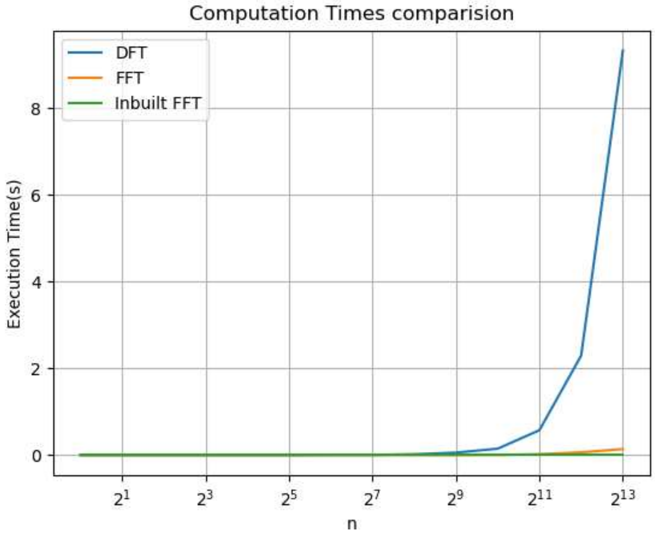
3.7. The following code plots magnitude and phase plots of $X(k)$, $H(k)$ and $Y(k)$.

```
https://github.com/yashwanthguguloth24/
    EE3025−DSP−lab/tree/main/
    Assignment1/codes/ee18btech11017_2.py
```



3.8. *Computation Times:*

We can compare the computation times for DFT,FFT and Inbuit-FFT algorithms for N = 2^M, for N = 1 (2^0) to 8192 (2^{13}).



From above plot we can observe that, computation time for DFT rises exponentially as we increase N in powers of 2 but FFT and Inbuilt-FFT are much faster.

3.9. The following code plots the above time comparision plot.

```
https://github.com/yashwanthguguloth24/
    EE3025−DSP−lab/tree/main/
    Assignment1/codes/ee18btech11017_3.py
```

3.10. Obtaining 8-Point FFT using DFT Matrix

$$\begin{bmatrix} X(0)\\ X(1)\\ X(2)\\ X(3)\\ X(4)\\ X(5)\\ X(6)\\ X(7) \end{bmatrix} = \begin{bmatrix} W_8^0 W_8^0 W_8^0 W_8^0 W_8^0 W_8^0 W_8^0 W_8^0\\ W_8^0 W_8^1 W_8^2 W_8^3 W_8^4 W_8^5 W_8^6 W_8^7\\ W_8^0 W_8^2 W_8^4 W_8^6 W_8^8 W_8^{10} W_8^{12} W_8^{14}\\ W_8^0 W_8^3 W_8^6 W_8^9 W_8^{12} W_8^{15} W_8^{18} W_8^{21}\\ W_8^0 W_8^4 W_8^8 W_8^{12} W_8^{16} W_8^{20} W_8^{24} W_8^{28}\\ W_8^0 W_8^5 W_8^{10} W_8^{15} W_8^{20} W_8^{25} W_8^{30} W_8^{35}\\ W_8^0 W_8^6 W_8^{12} W_8^{18} W_8^{24} W_8^{30} W_8^{36} W_8^{42}\\ W_8^0 W_8^7 W_8^{14} W_8^{21} W_8^{28} W_8^{35} W_8^{42} W_8^{49} \end{bmatrix} \begin{bmatrix} 1\\ 2\\ 3\\ 4\\ 2\\ 1\\ 0\\ 0 \end{bmatrix} \tag{3.10.1}$$

$$\implies \begin{bmatrix} X(0)\\ X(1)\\ X(2)\\ X(3)\\ X(4)\\ X(5)\\ X(6)\\ X(7) \end{bmatrix} = \begin{bmatrix} 13\\ -3.121-6.535j\\ 1j\\ 1.121-0.535j\\ -1\\ 1.121+0.535j\\ -1j\\ -3.121+6.535j \end{bmatrix} \tag{3.10.2}$$

Similarly,

$$\begin{bmatrix} H(0)\\ H(1)\\ H(2)\\ H(3)\\ H(4)\\ H(5)\\ H(6)\\ H(7) \end{bmatrix} = \begin{bmatrix} W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0 & W_8^0\\ W_8^0 & W_8^1 & W_8^2 & W_8^3 & W_8^4 & W_8^5 & W_8^6 & W_8^7\\ W_8^0 & W_8^2 & W_8^4 & W_8^6 & W_8^8 & W_8^{10} & W_8^{12} & W_8^{14}\\ W_8^0 & W_8^3 & W_8^6 & W_8^9 & W_8^{12} & W_8^{15} & W_8^{18} & W_8^{21}\\ W_8^0 & W_8^4 & W_8^8 & W_8^{12} & W_8^{16} & W_8^{20} & W_8^{24} & W_8^{28}\\ W_8^0 & W_8^5 & W_8^{10} & W_8^{15} & W_8^{20} & W_8^{25} & W_8^{30} & W_8^{35}\\ W_8^0 & W_8^6 & W_8^{12} & W_8^{18} & W_8^{24} & W_8^{30} & W_8^{36} & W_8^{42}\\ W_8^0 & W_8^7 & W_8^{14} & W_8^{21} & W_8^{28} & W_8^{35} & W_8^{42} & W_8^{49} \end{bmatrix} \begin{bmatrix} 1\\ -0.5\\ 1.25\\ -0.65\\ 0.3125\\ -0.15625\\ 0.078125\\ -0.0390625 \end{bmatrix} \tag{3.10.3}$$

$$\implies \begin{bmatrix} H(0)\\ H(1)\\ H(2)\\ H(3)\\ H(4)\\ H(5)\\ H(6)\\ H(7) \end{bmatrix} = \begin{bmatrix} 1.32\\ 0.858-0.514j\\ -0.015-0.007j\\ 0.516+1.829j\\ 3.96\\ 0.516-1.829j\\ -0.015+0.007j\\ 0.858+0.514j \end{bmatrix} \tag{3.10.4}$$

So,

$$\begin{bmatrix} Y(0)\\ Y(1)\\ Y(2)\\ Y(3)\\ Y(4)\\ Y(5)\\ Y(6)\\ Y(7) \end{bmatrix} = \begin{bmatrix} X(0)\cdot H(0)\\ X(1)\cdot H(1)\\ X(2)\cdot H(2)\\ X(3)\cdot H(3)\\ X(4)\cdot H(4)\\ X(5)\cdot H(5)\\ X(6)\cdot H(6)\\ X(7)\cdot H(7) \end{bmatrix} \tag{3.10.5}$$

$$\implies \begin{bmatrix} Y(0)\\ Y(1)\\ Y(2)\\ Y(3)\\ Y(4)\\ Y(5)\\ Y(6)\\ Y(7) \end{bmatrix} = \begin{bmatrix} 17.16\\ -6.04-4j\\ -0.007-0.015j\\ 1.55+1.77j\\ -3.96\\ 1.55-1.77j\\ 0.007+0.015j\\ -6.04+4j \end{bmatrix} \tag{3.10.6}$$

3.11. The following is the C program to compute the N-point FFT (where N is of the form 2^n)

```
https://github.com/yashwanthguguloth24/
     EE3025−DSP−lab/tree/main/
     Assignment1/codes/
     ee18btech11017_C_impl.c
```