Experiment No: 6 Date: 01/10/24

# DFT AND IDFT

## Aim:

1.DFT using inbuilt function and without using inbuilt function. Also plot magnitude and phase plot of DFT

2.IDFT using inbuilt function and without using inbuilt function.

## Theory:

### Discrete Fourier Transform (DFT)

The **Discrete Fourier Transform (DFT)** is a mathematical transformation used to analyze the frequency content of discrete signals. For a sequence x[n] of length N, the DFT is defined as:

$$X[k] = \sum_{n=0}^{N} x[n] \cdot e^{-j\frac{2\pi}{N}nk}, \; k = 0, 1, 2, \ldots, N-1$$

- X[k] is the DFT of the sequence x[n].
- The exponential factor represents $e^{-j\frac{2\pi}{N}nk}$ the complex sinusoidal basis functions.
- The DFT maps the time-domain signal into the frequency domain.

### Inverse Discrete Fourier Transform (IDFT)Method:

The **Inverse Discrete Fourier Transform (IDFT)** is used to convert a frequency-domain sequence X[k] back into its time-domain sequence x[n]. The IDFT is defined as:

$$x[n] = \frac{1}{N}\sum_{k=0}^{N} X[k] \cdot e^{j\frac{2\pi}{N}nk}, \; n = 0, 1, 2, \ldots, N-1$$

- The IDFT takes the frequency components X[k] and reconstructs the original sequence x[n].
- The exponential factor $e^{j\frac{2\pi}{N}nk}$ is the inverse of the DFT's complex sinusoidal basis functions

.

### Application

- Spectrum (Analysis)
- Filtering
- Compression
- Modulation
- Convolution
- Demodulation

- Equalization
- Restoration
- Detection
- Estimation

## Program:

### 1. Discrete Fourier Transform (DFT)

```
clc;
clear all;
close all;
x=input("enter sequence:");
N=input("enter the N point:");
l=length(x);
x=[x zeros(1,N-l)];
X=zeros(1,N);
for k=0:N-1
    for n=0:N-1
        X(k+1)=X(k+1)+x(n+1)*exp(-1j*2*pi*n*k/N);
    end
end
disp('X');
disp(X);
disp('round(X)');
disp(round(X));
%verification
disp('fft');
disp(fft(x));

k=0:N-1;
magX=abs(X);
```

```
phaseX=angle(X);
subplot(2,1,1);
stem(k,magX);
title("Magnitude Plot");
hold on;
plot(k,magX);
subplot(2,1,2);
stem(k,phaseX);
hold on;
title("Phase Plot");
plot(k,phaseX);
```

**2.IDFT**

```
clc;
clear all;
close all;
X=input("enter sequence:");
N=input("enter the n point:");
l=length(X);
X=[X zeros(1,N-l)];
x=zeros(N,1);
for k=0:N-1
    for n=0:N-1
        x(n+1)=x(n+1)+X(k+1)*exp(1j*2*pi*n*k/N);
    end
end
x=1/N.*x;
disp('x');
disp(x);
disp('round(x)');
```

```
disp(round(x));
disp('ifft');
disp(ifft(X));
```

## Result:

Performed

1)DFT using inbuilt function and without using inbuilt function. Also plotted magnitude and phase plot of DFT.

2)IDFT using inbuilt function and without using inbuilt function.

and verified the result.

## Observation:

**1.DFT**

enter sequence:[1 1 1 0]

enter the N point:8

X

Columns 1 through 7

3.0000 + 0.0000i 1.7071 - 1.7071i 0.0000 - 1.0000i 0.2929 + 0.2929i 1.0000 + 0.0000i 0.2929 - 0.2929i -0.0000 + 1.0000i

Column 8

1.7071 + 1.7071i

round(X)

Columns 1 through 7

3.0000 + 0.0000i 2.0000 - 2.0000i 0.0000 - 1.0000i 0.0000 + 0.0000i 1.0000 + 0.0000i 0.0000 + 0.0000i 0.0000 + 1.0000i

Column 8

2.0000 + 2.0000i

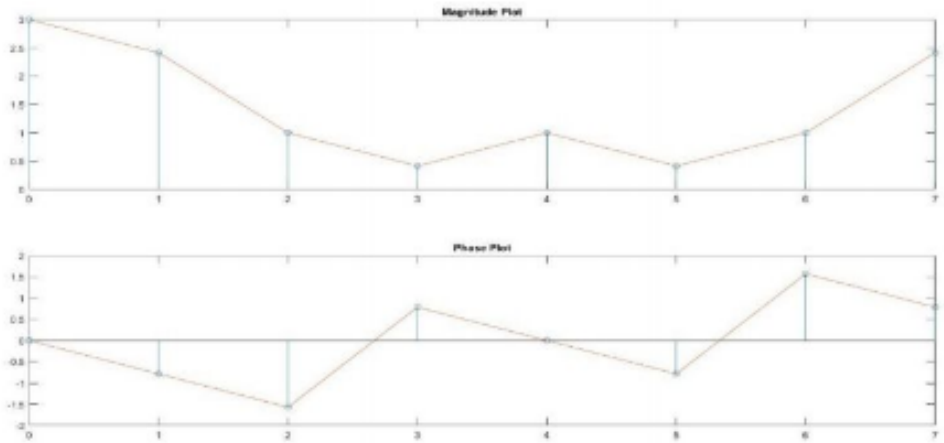
fft

Columns 1 through 7

3.0000 + 0.0000i 1.7071 - 1.7071i 0.0000 - 1.0000i 0.2929 + 0.2929i 1.0000 + 0.0000i 0.2929 - 0.2929i 0.0000 + 1.0000i

Column 8

1.7071 + 1.7071i



**2.IDFT**

```
enter sequence: [3.0000 + 0.0000i   1.7071 - 1.7071i   0.0000 - 1.0000i   0.2929 + 0.2929i
1.0000 + 0.0000i   0.2929 - 0.2929i   0.0000 + 1.0000i]
enter the n point:8
x
   0.7866 - 0.2134i
   0.6982 - 0.0000i
   0.7866 + 0.2134i
  -0.0000 + 0.3018i
   0.2134 + 0.2134i
   0.3018 - 0.0000i
   0.2134 - 0.2134i
   0.0000 - 0.3018i

round(x)
     1
     1
     1
     0
     0
     0
     0
     0

ifft
  Columns 1 through 7

   0.7866 - 0.2134i   0.6982 + 0.0000i   0.7866 + 0.2134i   0.0000 + 0.3018i   0.2134 +
0.2134i   0.3018 + 0.0000i   0.2134 - 0.2134i

  Column 8

   0.0000 - 0.3018i
```