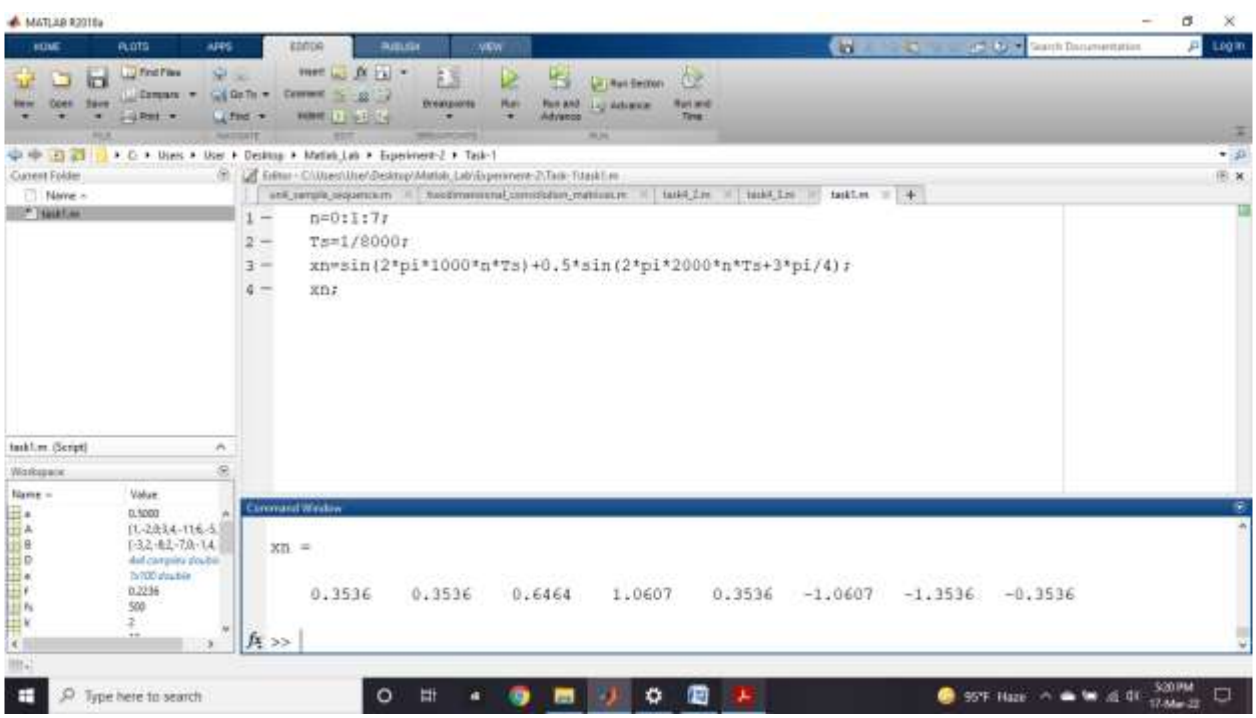
Task-1: Consider a continuous time signal, x(t) = $\sin(2\pi 2000t) + \frac{1}{2}\sin(2\,\pi 200t + \frac{3\pi}{4})$
Determine sampled signal *x(n.Ts) = x(n)*, using matlab, taking *N* = 8 samples at sampling
rate, *Fs* = 8000Hz (samples/sec).
The sampled signal,
xs(n) = x(nTs) = $\sin(2\pi 2000nTs) + \frac{1}{2}\sin(2\,\pi 200nTs + \frac{3\pi}{4})$; Where Ts = $\frac{1}{Fs} = \frac{1}{8000}$sec is the sampling period. Let us use matlab code to determine sampled sequence *x(n)*.

Solution:

```
n=0:1:7;
Ts=1/8000;
xn=sin(2*pi*1000*n*Ts)+0.5*sin(2*pi*2000*n*Ts+3*pi/4);
xn;
```

Output:



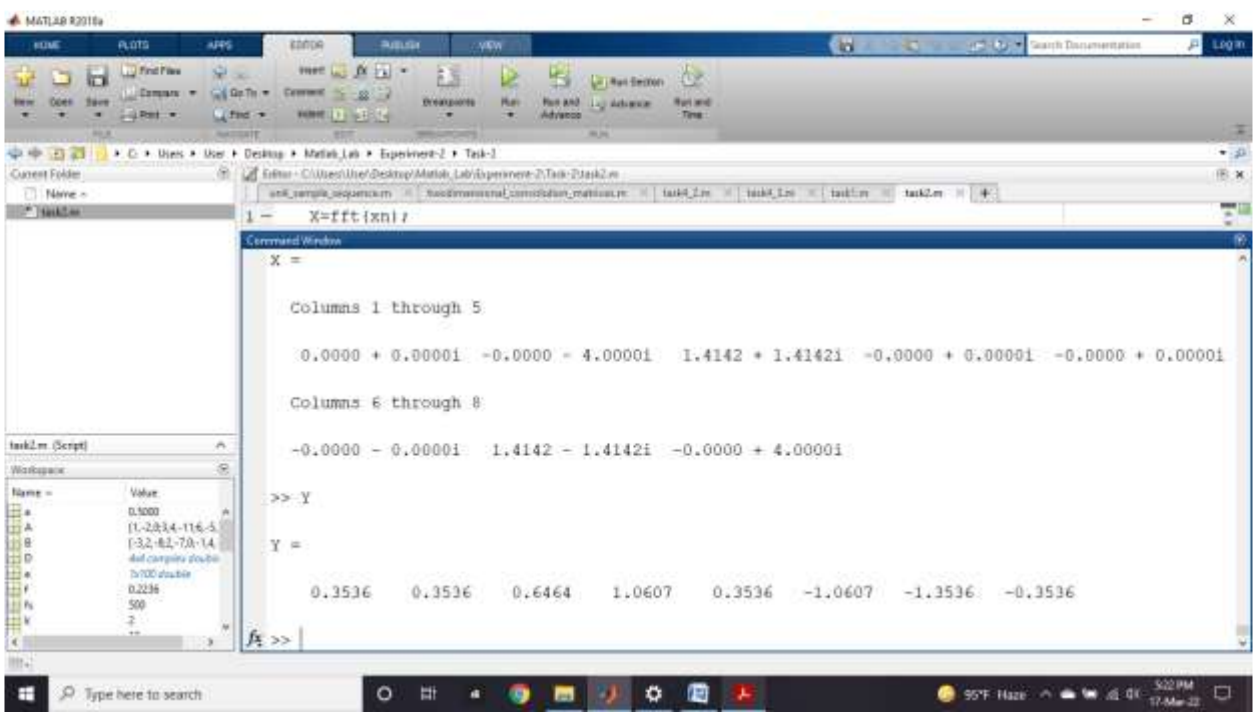
Observation: This is a continuous time signal x(t). After implementation of this signal it represents the sampled signal xn. In where x-axis represents n samples and y-axis represents x(n) signal.

Task-2: Determine DFT of above sequence using Matlab.

Solution:

```
X=fft(xn);
%IDFT of above sequence will retrieve the original
sequence x(n) like.
Y=ifft(X);
```

Output:



Observation: This is a continuous time signal x(t). After implementation of this signal it represents how inverse discrete time fourier transformation(IDFT) is worked. In where i represents the imaginary number.

Task-3: If *x(n)* ↔ *X(m);* both *x(n)* and *X(m)* are the vector of length *N* then their relation can be expressed in a different way like,

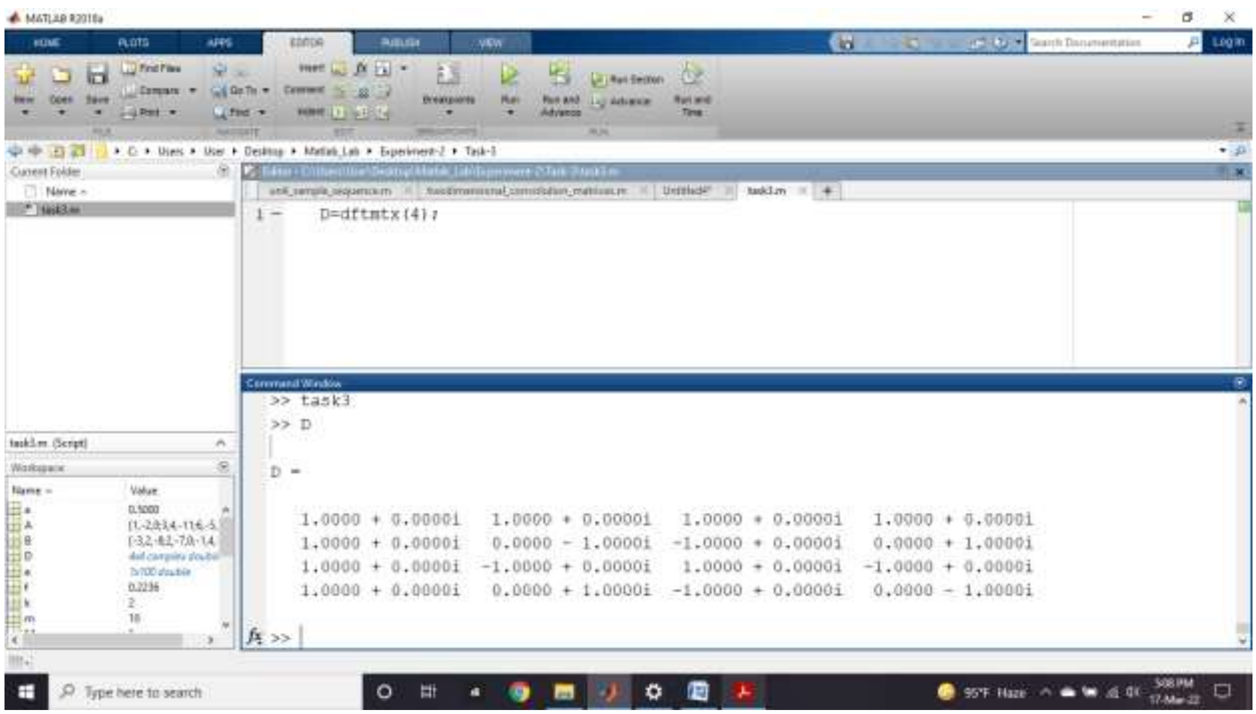
X = **D**N *x*
Where ***x*** *= [x(0) x(1) x(2) … … … x(N-1)]T* ***,***
X = [X(0) X(1) X(2) … … … X(N-1)]T ,

Determine the **D**N matrix of dimension of 4×4 using Matlab.

Solution:

D=dftmtx(4)

Output:



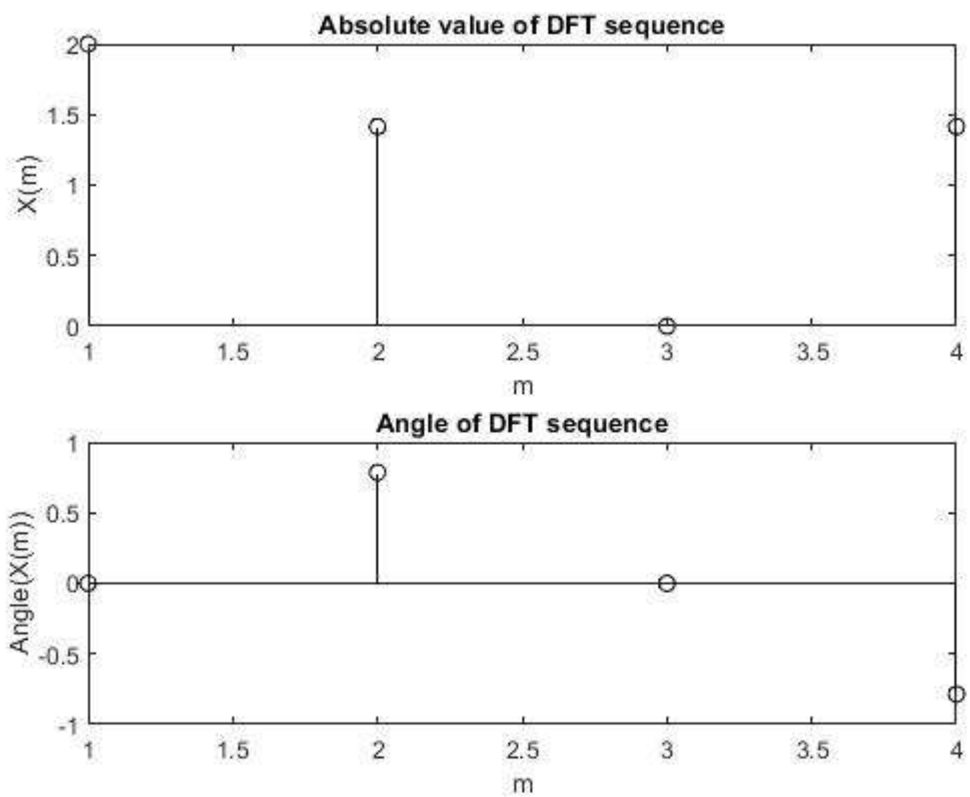
Observation: This is a continuous time signal x(t). After implementation of this signal it represents the **D**N matrix of dimension of 4×4. In where i represents the imaginary number.

Task-4.1: Write Matlab code to determine DFT of *x (n)* = {1, 0, 0, 1} hence show the plot of
X(m).

Solution:

```
x=[1 0 0 1];
y=fft(x);
subplot(2,1,1)
stem(abs(y), 'k')
xlabel('m')
ylabel('X(m)')
title('Absolute value of DFT sequence')
subplot(2,1,2)
stem(angle(y), 'k')
xlabel('m')
ylabel('Angle(X(m))')
title('Angle of DFT sequence')
```

Output:



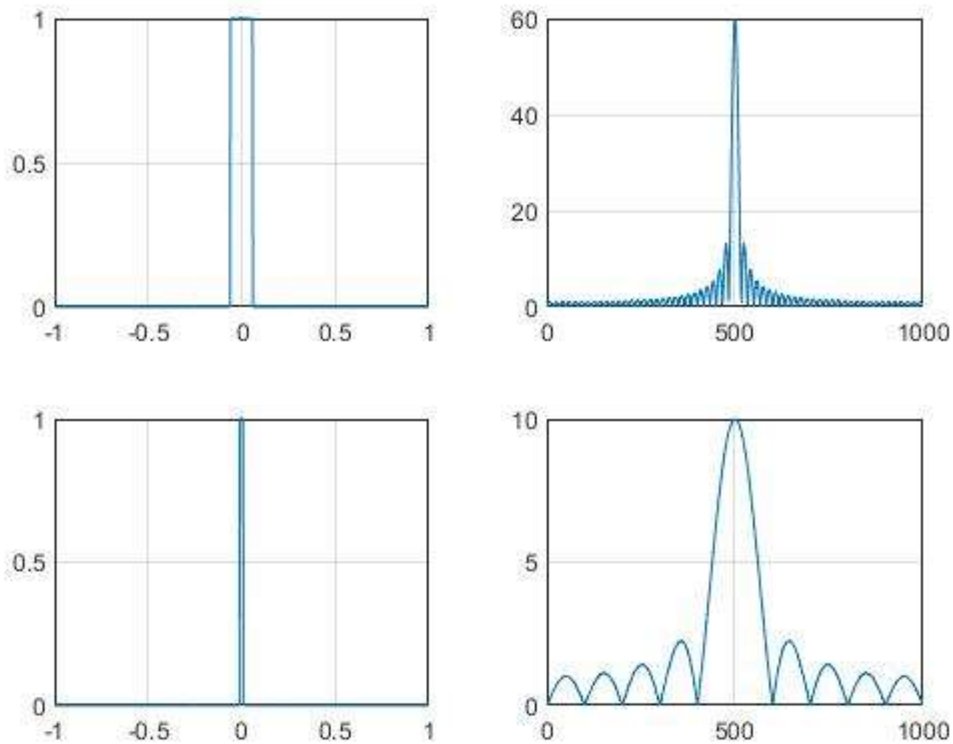
Observation: This is a continuous time signal x(t). After implementation of this signal it represents the absolute value of DFT and the angles of DFT. In where x-axis represents m samples and y-axis represents x(m) signal.

Task-4.2: DTFT on a rectangular pulse.

Solution:

```
fs = 500;
t = -1:1/fs:1;

x = rectpuls(t,0.12);
subplot(2,2,1)

plot(t, x)
grid on

y = fft(x);
y = fftshift(y);

subplot(2,2,2)
plot( abs(y))

grid on
x = rectpuls(t,0.02);

subplot(2,2,3)
plot(t, x)

grid on
y = fft(x);

y = fftshift(y);
subplot(2,2,4)

plot( abs(y))
grid on
```

Output:



Observation: This is a continuous time signal x (t). After implementation of this signal, it represents the DTFT on a rectangular pulse. In where the x-axis represents m samples and the y-axis represents x (m) signal.

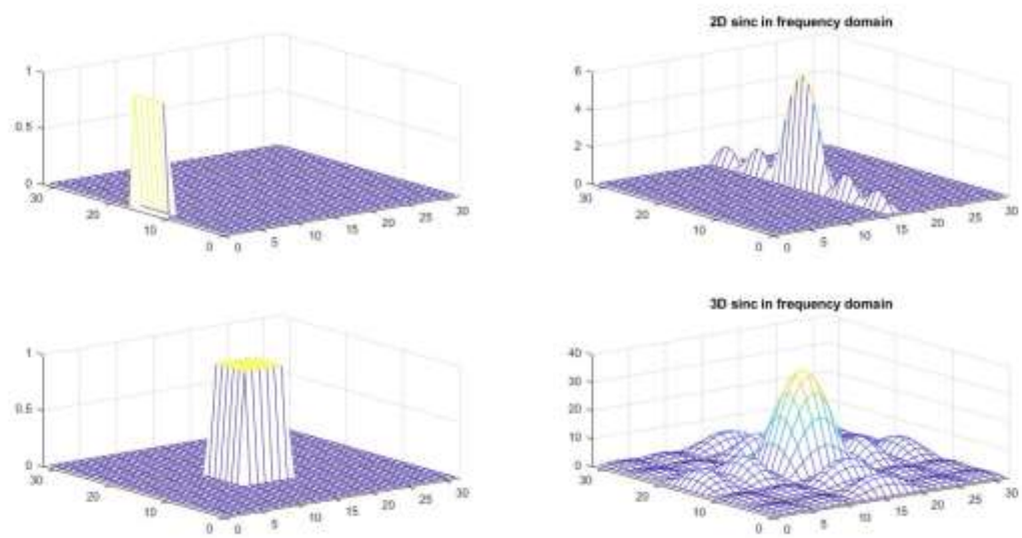
Task-4.3: DTFT on a 3D rectangular pulse.

Solution:

```
x=zeros(32);
x(12:17)=ones(6,1);
subplot(2,2,1)
title('2D rectangular pulse')
mesh(x)
x=fft(x);
x=fftshift(x);%Dc avlue is at the corner of the array
%let us move it at the middle
```

```
subplot(2,2,2)
mesh(abs(x))
title('2D sinc in frequency domain')
x=zeros(32);
x(12:17,12:17)=ones(6);
subplot(2,2,3)
title('3D rectangular pulse')
mesh(x)
x=fft2(x);
x=fftshift(x);%Dc avlue is at the corner of the array
%let us move it at the middle
subplot(2,2,4)
mesh(abs(x))
title('3D sinc in frequency domain')
```

Output:



Observation: This is a continuous time signal x (t). After implementation of this signal, it represents the DTFT on a 3D rectangular pulse. In where the x-axis represents m samples and the y-axis represents x (m) signal.