DSAA COMPUTER

ASSIGNMENT-3

NAME :- ANIRUDH KANNAN V P

ROLL NO: - 201601004

QUESTION 1:-

1. Given a signal x(t) that is not strictly band limited. Devise a method for finding the sampling rate.

An example:

1. Generate a signal

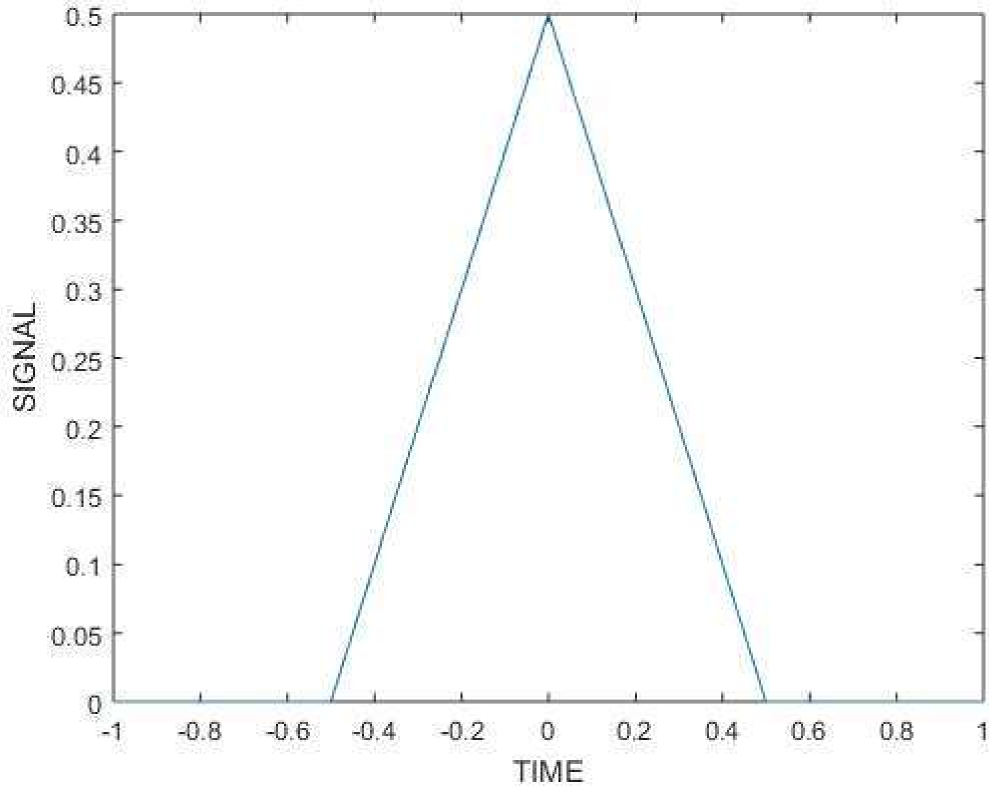
$x(t) = t + T$ for $-T < t < 0$

$= T - t$ for $0 < t < T$

MATLAB CODE:-

```
%%COMP ASSIGN-2
%%ANIRUDH KANNAN V P
%%201601004
%%UG2 CSE

clc
clear all
close all

%time vector
T=0.5;
timevec=-2*T:0.05:2*T;
t=-10:1:10;

%Fs
Fs=20;

%TO PLOT x(t+T) and x (T-t)
xt=zeros(size(timevec));
xt(timevec>=-1*T & timevec<=0)=max(0,(timevec(timevec>=0 &
timevec<=T )));
xt(timevec>=0 & timevec<=T)=T-max(0,(timevec(timevec>=0 &
timevec<=T )));
figure(1);
plot(timevec,xt);
xlabel('TIME');
ylabel('SIGNAL');
```

OUTPUT AND DISCUSSION:-



The main motive is to plot the signal:

x(t) = t + T   for -T < t < 0

= T-t   for 0 < t < T

So i have created a timevector from -1 to 1 with spacing of 0.05.First i have created a zero vector so the value of the signal is zero everywhere.

Then i have created th signal between -1 and 1 by

```
xt(timevec>=-1*T & timevec<=
0)=max(0,(timevec(timevec>=0 & timevec<=T)));

xt(timevec>=0 & timevec<=T)=T-
max(0,(timevec(timevec>=0 & timevec<=T )));
```

Creating left side and right side of the signal seperately and storing it in xt. T was given as 0.5.

Then i have plotted the signal using plot function.

QUESTION 2 AND 3:-

2.Determine its fourier transform using the matlab function fft

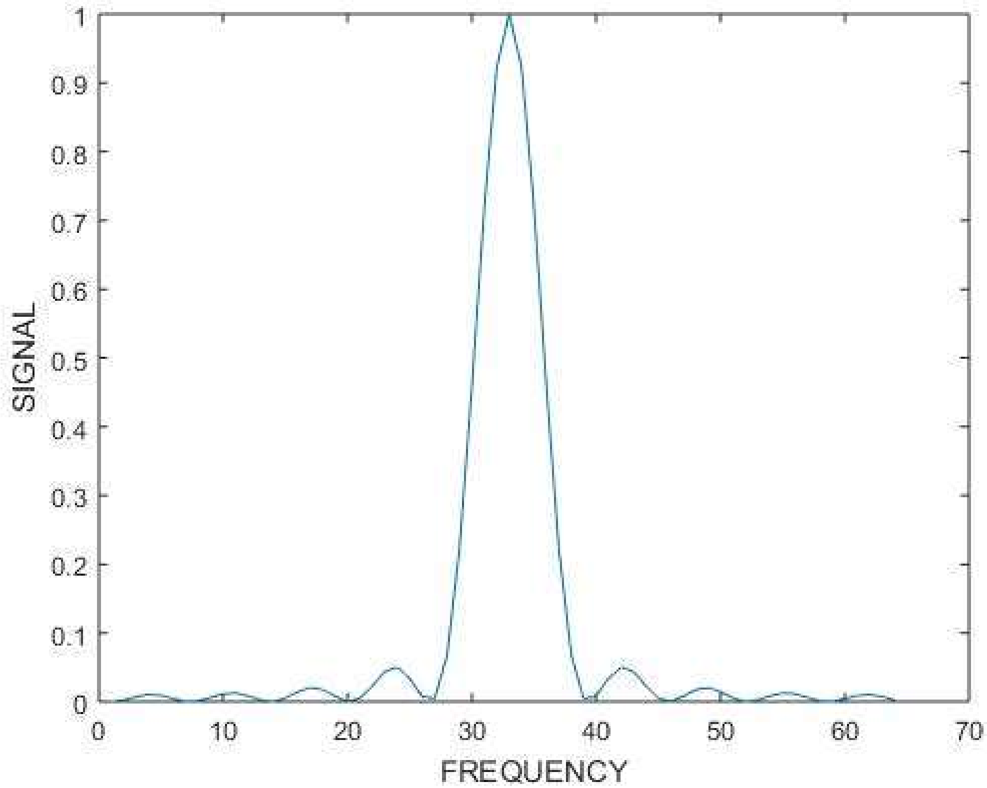
3.Plot the fourier transform X(w) versus frequency and verify it is band limited or not

MATLAB CODE:-

```
%%COMP ASSIGN-2
%%ANIRUDH KANNAN V P
%%201601004
%%UG2 CSE

clc
clear all
close all

%time vector
T=0.5;
timevec=-2*T:0.05:2*T;
t=-10:1:10;

%Fs
Fs=20;

%TO PLOT x(t+T) and x (T-t)
xt=zeros(size(timevec));
xt(timevec>=-1*T & timevec<=0)=max(0,(timevec(timevec>=0 & timevec<=
0.5 )));
xt(timevec>=0 & timevec<=T)=T-max(0,(timevec(timevec>=0 & timevec<=
```

```
0.5 )));


%fft of new x(t)

%changing to nearest power of 2
nearest = nextpow2(length(xt));

nfft=2^(nearest);

%FINDING THE FOURIER TRANSFORM AND FFTSHIFT
xw=fft(xt,nfft);
xw=fftshift(xw);

fvec=Fs/2*linspace(-1,1,nfft);

figure(1);


xw=xw/max(xw);

plot(abs(xw));
xlabel('FREQUENCY');
ylabel('SIGNAL');
```

OUTPUT AND DISCUSSION:-



HERE THE SIGNAL GIVEN BY x(t) is fourier transformed my using the fft() function fast fourier transform function.

```
nearest = nextpow2(length(xt));

nfft=2^(nearest);
```

is calculated for fourier transform.Then the function is fourier transransformed using fft() function and shifted using ffttransform()and plotted using plot function.

COMMANDS USED:-

```
xw=fft(xt,nfft);
xw=fftshift(xw);

fvec=Fs/2*linspace(-1,1,nfft);

figure(1);
```

```
xw=xw/max(xw);

plot(abs(xw));
xlabel('FREQUENCY');
ylabel('SIGNAL');
```

From the figure it is evident that the signal does not have a bandwith and it extends upto infinity, and the energy can be found using parvesal's energy relation.

QUESTION 4 AND 5 AND 6:-

4. Now define a new way of deciding the bandwidth of the signal.

For example: the spectral width in which most of the energy resides. ("most of the" means 99%)

5. Next, based on this new bandwidth define the sampling rate

6. Evaluate the fourier transform based on theoretical definition.

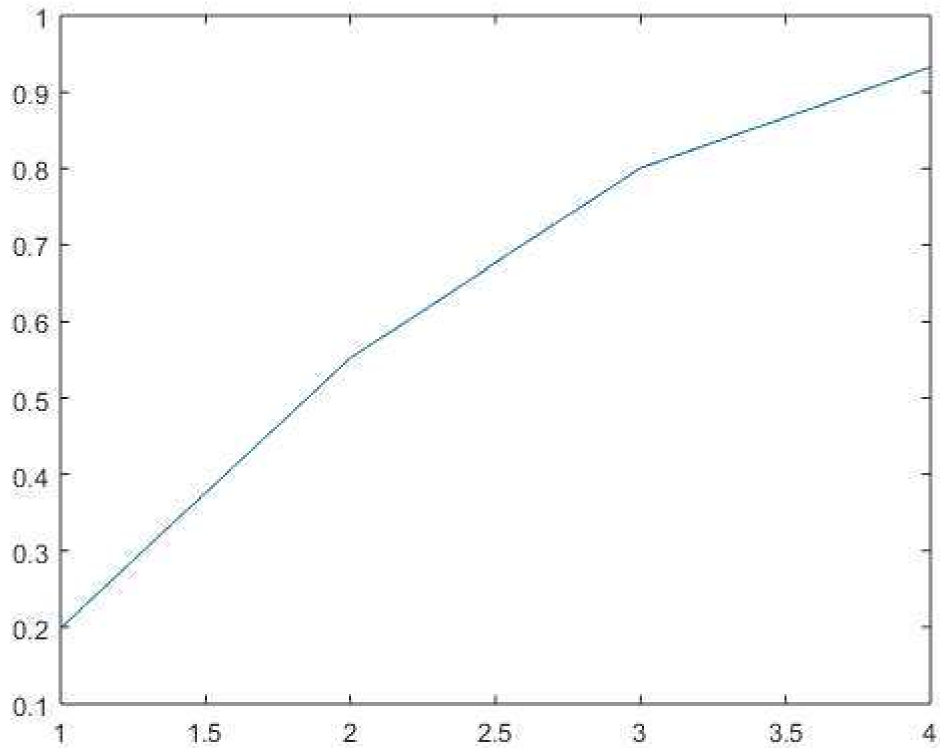
Compute the energies using the integral formula for energy and determine the location at which the partial energy is 99%.

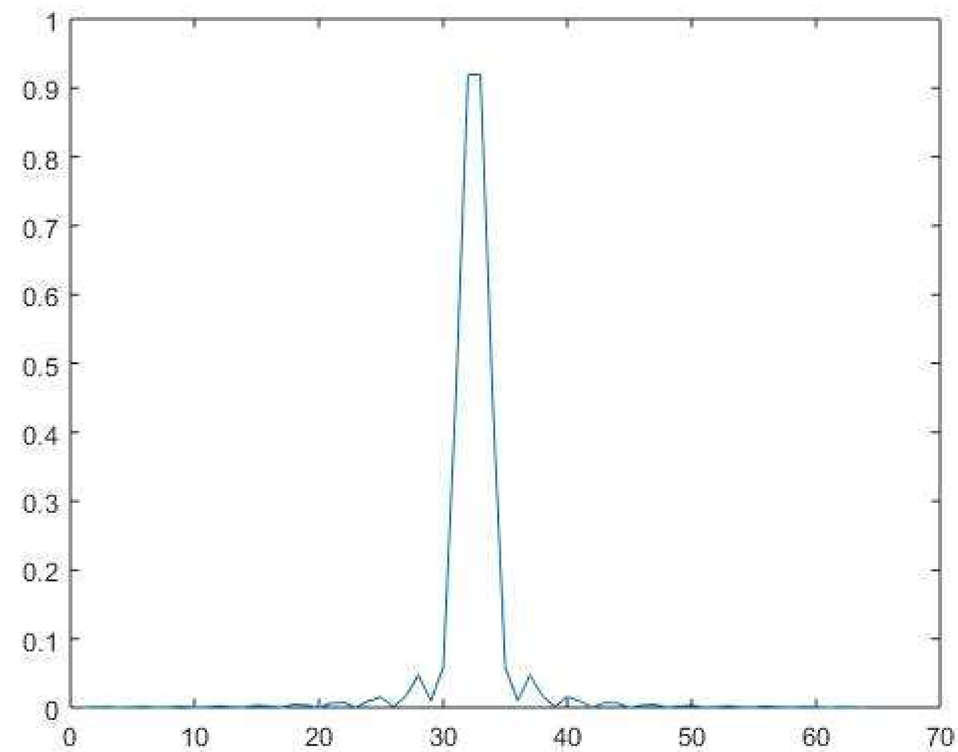
MATLAB CODE:-

```
%%COMP ASSIGN-2
%%ANIRUDH KANNAN V P
%%201601004
%%UG2 CSE

clc
clear all
close all

%time vector
T=0.5;
timevec=-2*T:0.05:2*T;
t=-10:1:10;

%Fs
Fs=20;

%TO PLOT x(t+T) and x (T-t)
xt=zeros(size(timevec));
xt(timevec>=-1*T & timevec<=0)=max(0,(timevec(timevec>=0 &
timevec<=T )));
xt(timevec>=0 & timevec<=T)=T-max(0,(timevec(timevec>=0 &
timevec<=T )));


%fft of new x(t)

%changing to nearest power of 2
nearest = nextpow2(length(xt));

nfft=2^(nearest);

%FINDING THE FOURIER TRANSFORM AND FFTSHIFT
xw=fft(xt,nfft);
xw=fftshift(xw);

fvec=Fs/2*linspace(-1,1,nfft);

figure(1);


xw=xw/max(xw);


%Energy of x(w)
energyundergraph = trapz(abs(xw).^2);
NFFT = 2^(nearest);
energynow=0;

%STARTING FROM MIDDLE POINT
startingpointnow = NFFT/2;
stopingpointnow = NFFT/2;

fmx = 1;
```

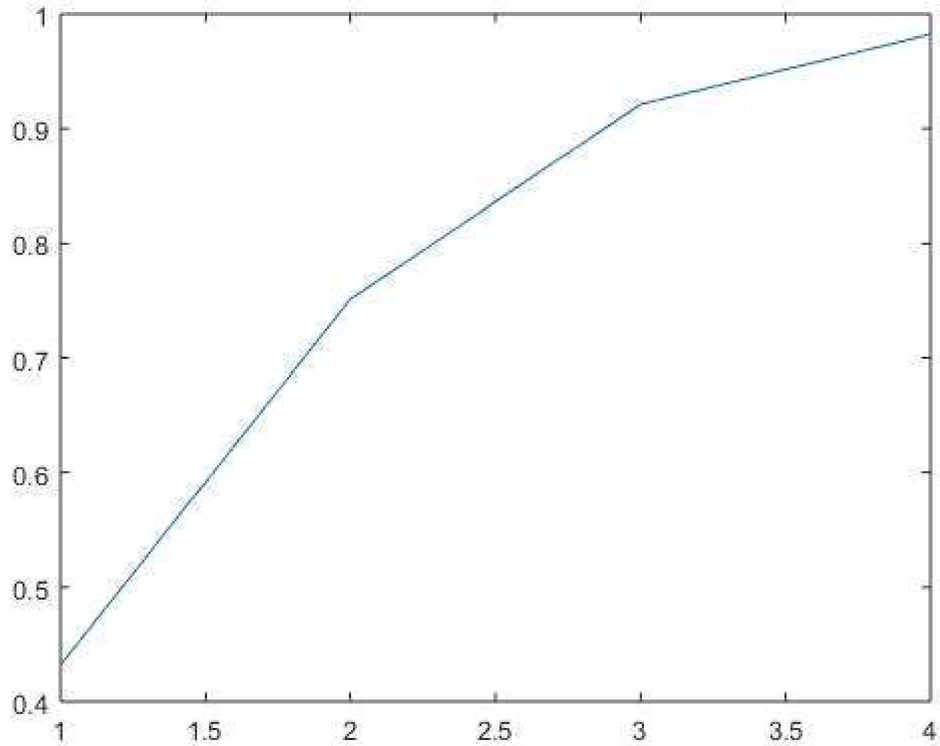
```
while energynow*100/energyundergraph <= 99 % 99PERCENT
energynow = sum(abs(xw(startingpointnow : stopingpointnow)).^2);

%GOING IN BOTH DIRECTIONS AND ADDING ENERGY.
startingpointnow = startingpointnow - 1;
stopingpointnow = stopingpointnow + 1;

egvec(fmx) = energynow/energyundergraph;

%ITERATING
fmx = fmx + 1;

end
disp(startingpointnow);
disp(stopingpointnow);
freq = Fs/2 * linspace(-1, 1, NFFT);
bandwidth = freq(ceil(stopingpointnow)) - freq(ceil(startingpointnow));
disp(bandwidth);
figure;
plot(egvec);
xlim([1 4]);

%using sinc()^2 function
sincfunction=(sin(T*2*pi*fvec)./(T*2*pi*fvec)).^2;
sincfunction(fvec==0)=1;
figure;
plot(abs(sincfunction));

%Finding energy using area under code
energyundergraph=trapz(sincfunction.^2);
f=0;
energynow=0;
delf=fvec(2)-fvec(1);

Nm=Fs/delf;

startingpointnow=Nm/2;
stopingpointnow=Nm/2;

while energynow*100/energyundergraph<=99
f=f+1;
energynow=trapz(abs(sincfunction(startingpointnow : stopingpointnow)).^
2);
startingpointnow = startingpointnow - 1;
stopingpointnow = stopingpointnow + 1;
egsinc(f)=energynow/energyundergraph;
end

disp(startingpointnow);
disp(stopingpointnow);
freq = Fs/2 * linspace(-1, 1, NFFT);
bandwidth = freq(ceil(stopingpointnow)) - freq(ceil(startingpointnow));
disp(bandwidth);
figure;
plot(egsinc);
xlim([1 4]);
```

OUTPUT AND DISCUSSION:-





1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0
10
20
30
40
50
60
70



To make the signal band limited we will first consider the signal upto 99 percent of energy . To find the signal of the signal upto 99 percent we first consider the middle most point. From there we move towards left as well as right, adding on to energy while it is 99 percent of the total energy. The code is as given below explained in each step:-

```
%Energy of x(w)
energyundergraph = trapz(abs(xw).^2);
NFFT = 2^(nearest);
energynow=0;

%STARTING FROM MIDDLE POINT
startingpointnow = NFFT/2;
```

```
stopingpointnow = NFFT/2;

fmx = 1;

while energynow*100/energyundergraph <= 99 % 99PERCENT
energynow = sum(abs(xw(startingpointnow : stopingpointnow)).^2);

%GOING IN BOTH DIRECTIONS AND ADDING ENERGY.
startingpointnow = startingpointnow - 1;
stopingpointnow = stopingpointnow + 1;
```

This can be considered as a band limited signal close to the original signal whose energy is 99 percent of the original signal.

The sampling rate is now defined as 2*maximum frequency given that the signal is band limited . This is how based on new bandwidth sampling rate is defined.

The fourier transform based on theoretical we get sinc2(w) as the fourier transform of the signal. The energy is calculated as the area under the graph . The energy using sinc function is calculated using the same way.

```
%using sinc()^2 function
sincfunction=(sin(T*2*pi*fvec)./(T*2*pi*fvec)).^2;
sincfunction(fvec==0)=1;
figure;
plot(abs(sincfunction));

%Finding energy using area under code
energyundergraph=trapz(sincfunction.^2);
f=0;
energynow=0;
delf=fvec(2)-fvec(1);

Nm=Fs/delf;

startingpointnow=Nm/2;
stopingpointnow=Nm/2;

while energynow*100/energyundergraph<=99
f=f+1;
energynow=trapz(abs(sincfunction(startingpointnow : stopingpointnow)).^
2);
startingpointnow = startingpointnow - 1;
stopingpointnow = stopingpointnow + 1;
egsinc(f)=energynow/energyundergraph;
end

disp(startingpointnow);
disp(stopingpointnow);
```

```
freq = Fs/2 * linspace(-1, 1, NFFT);
bandwidth = freq(ceil(stopingpointnow)) - freq(ceil(startingpointnow));
disp(bandwidth);
figure;
plot(egsinc);
xlim([1 4]);
```

Thus the bandwidth values from both the methods agree. Thus from the figure it is evident that the partial energy is 99% around 0.9666 .

Both the graphs are given above and it can be clearly seen that the energies are almost equal. The slight errors or differences in energies are due to the fact that the energies are calculated in different ways.