Certificate of Originality

It is to certify, that the assignment for Signals and Systems submitted by **Umar Hayyat, Reg#: 2019-EE-360** is my original work. This work is free from any plagiarism, if copying of this work is detected I should be awarded **zero** marks.

Signature Umar Hayyat

<u>Problem 1:</u>

Let u[n] and v[n] are two signals and m = length (u) and n = length (v). Then w is the vector of length (m+n-1) whose kth element is

$$w(k) = \sum_{j} u(j)v(k-j+1)$$

MATLAB index cannot be zero. Hence, there is +1 in the formula i.e. for j=1, we will get the first term of output.

When m = n, this gives

w(1) = u(1)*v(1)
w(2) = u(1)*v(2)+u(2)*v(1)
w(3) = u(1)*v(3)+u(2)*v(2)+u(3)*v(1)
...
w(n) = u(1)*v(n)+u(2)*v(n-1)+ ... +u(n)*v(1)

Input function: $x[n] = n;$ where *n=0:R* *R=your registration number*

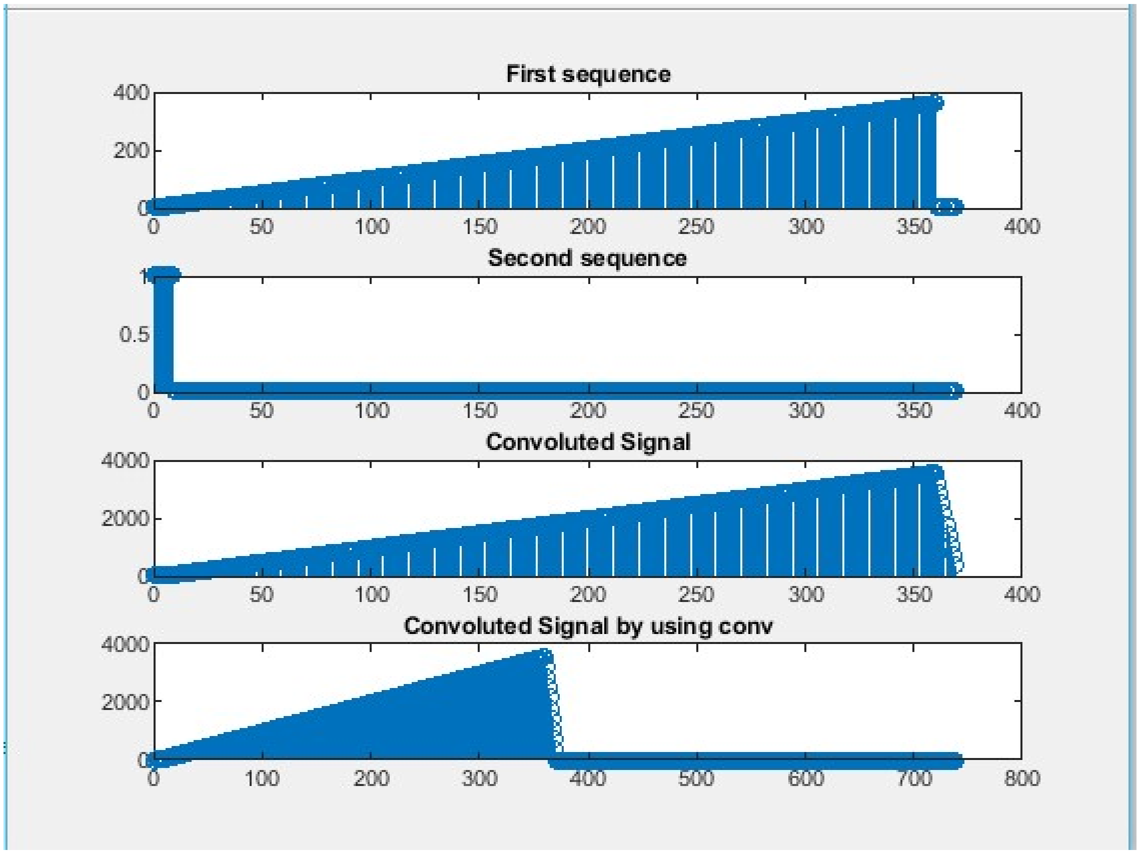
Impulse response: $h[n] = ones(1,10);$

- **Convolute** x[n] with h[n] using for loop and plot x[n], h[n] and y[n] in same window using subplot.
- Also verify your answer using conv command, also plot the output using conv command in the same window

Code:

```
n=0:360;
x=n;
h=ones(1,10);

n1=length(x);                %Find the length of a signal
n2=length(h);

N=n1+n2-1;                    %find the length of y(n)
x=[x,zeros(1,N-n1)];          %zero padding to make the length=N
h=[h,zeros(1,N-n2)];
y=zeros(1, N);               %Initialize the output with zero
%perform linear convolution
for n=1:N
    for k=1:n
        y(n)=y(n)+x(k)*h(n-k+1);
    end
end
disp(y);
%plot the inputs and outputs.
ny=0:N-1;
subplot(4,1,1);
stem (ny,x);
title ('First sequence');

subplot (4, 1, 2);
stem (ny,h);
title('Second sequence');

subplot (4, 1, 3);
stem (ny,y);
title('Convoluted Signal');

a=conv(x,h);                  % Convolutio by using conv command
subplot (4, 1, 4);
stem (a);
title('Convoluted Signal by using conv');
```

Waveform:



Problem 3:

One of the applications of signal compression and expansion is to fast forward/slow forward of sounds. Expansion of signal slows down the playing of a sound and compression of signal makes signal play fast.

Design an LTI system to compress/expand the time of a sound file. Show code and waveforms.

Hints:

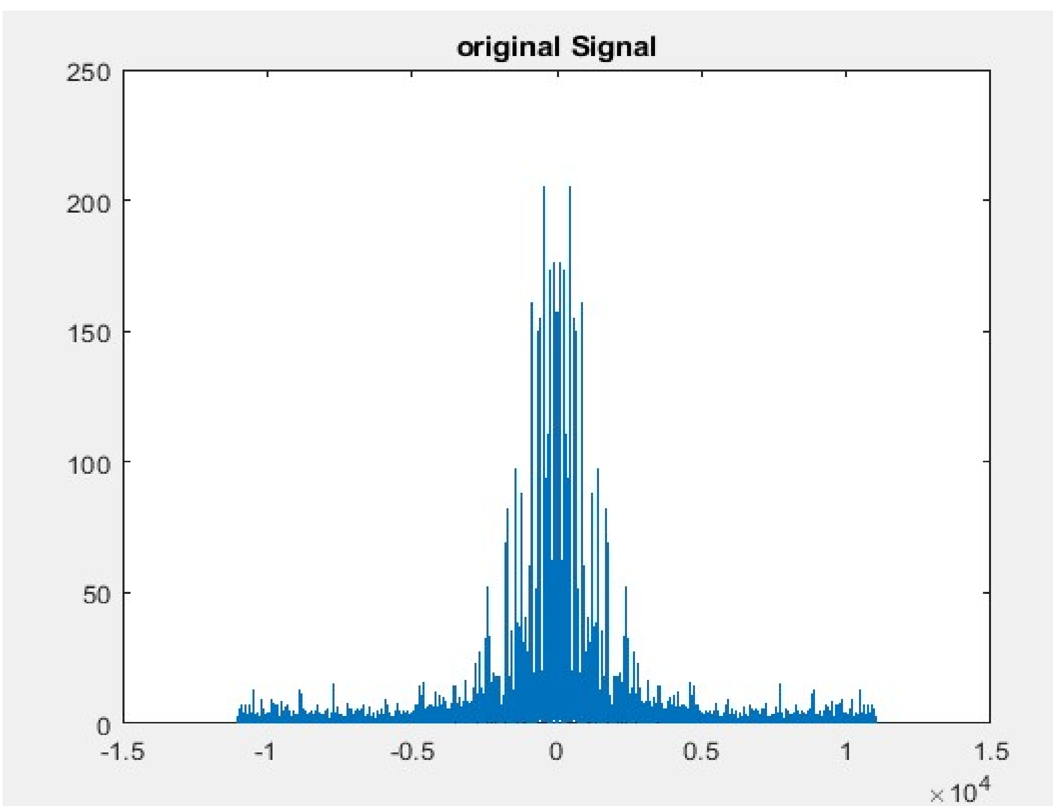
*Check **audioload** and **sound** commands in MATLAB for audio file*

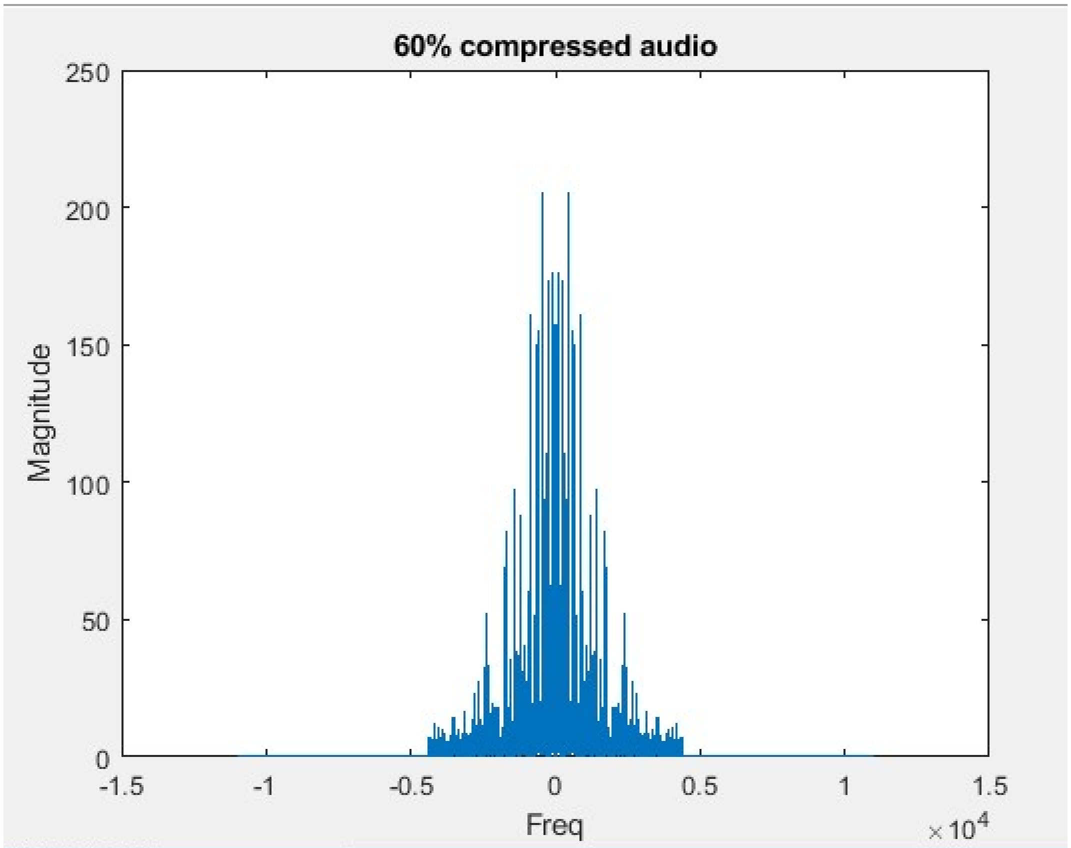
Compression mean increasing the speed by increasing the sampling frequency.

Code:

```
[x,fs]=audioread('Original.wav');     %read audio file
N=length(x);
vlcplayer=audioplayer(x,fs);
vlcplayer.play

t=fft(x,N);
X=fftshift(t);
f=-fs/2:fs/N:(fs/2-fs/N);        %frequency boundry
figure(1)
plot(f,abs(X))
title ('original Signal')

Xr=zeros(1,N);
Xr((N*((60/100)/2))+1:N*(1-(60/100)/2))=X((N*((60/100)/2))+1:N*(1-
(60/100)/2));
figure(2)
plot(f,abs((Xr)));
xr=real(ifft(fftshift(Xr)));
audiowrite('60%compressed.wav',xr,fs);
title('60% compressed audio')
xlabel('Freq'); ylabel('Magnitude');
```

Waveform:





Problem 4:

You are required to implement below shift system in MATLAB, t_0 is the amount of shift towards left or right.

$$y(t) = x(t - t_0).$$

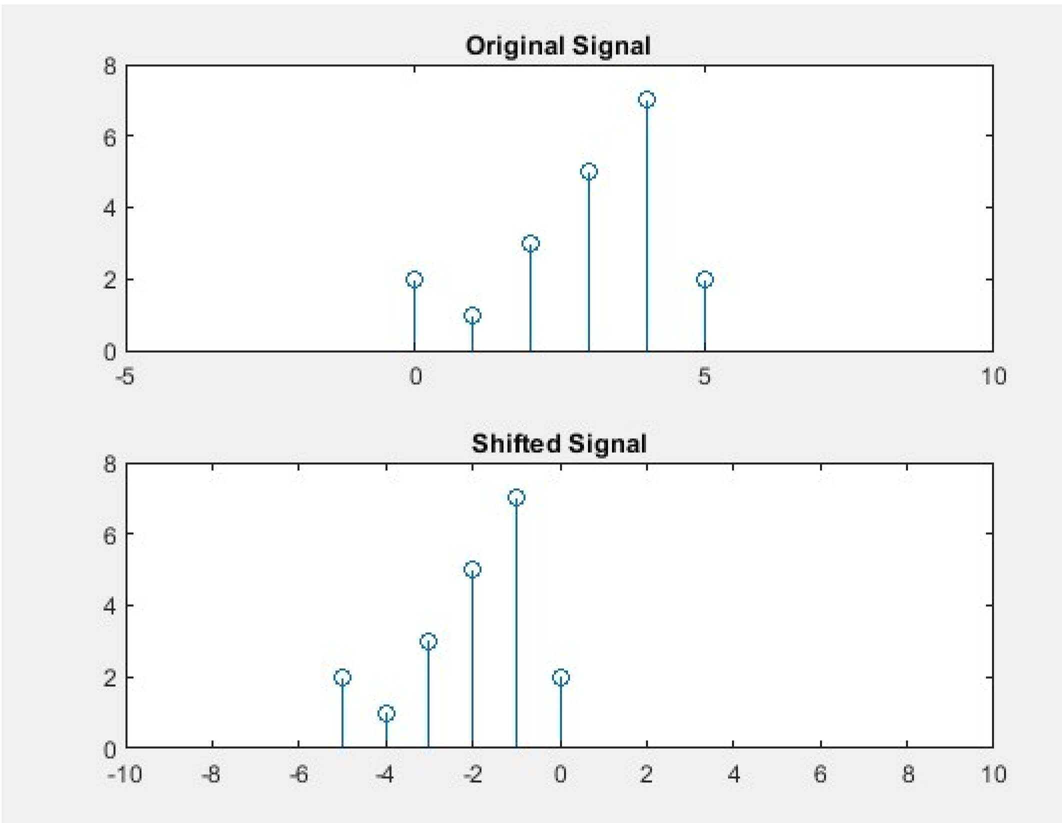
Implement MATLAB code, where shift should be taken as an input from the user. Show code and waveforms.

Code:

```
%Implementation of y(t)=X(t-to)

t=0:5;
x=[2 1 3 5 7 2];

subplot(2,1,1);
stem(t,x);                                          % plot in discrete
sequence
xlim([-5 10]);
title('Original Signal');

to=input('Enter a Number to shift signal: ');     %Get the valhue of to  from
user
if to>0
    a=-to;
    b=0;
elseif to<0
    a=0;
    b=-to;
else
    a=0;
    b=0;
end

subplot(2,1,2);
stem(t-to,x);
xlim([a-5 b+10]);
title('Shifted Signal');
```

Waveform:

```
>> Problem4
Enter a Number to shift signal: 5
>>
```



Conclusion:

I have learnt how we use Matlab to perform different tasks. I also learnt that if we do not have the information of any function that how it use or what is its syntax then we can use help command. I also observed that if we do not know the name of any command then we can use lookfor command.