ECE 311 final lab report

Name: Yang Shi, NetID: yangshi5

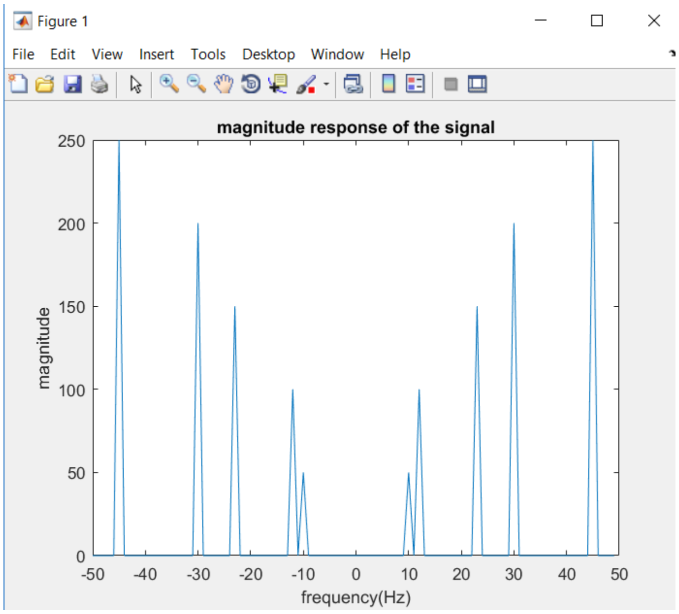
Question 1:

Code part:

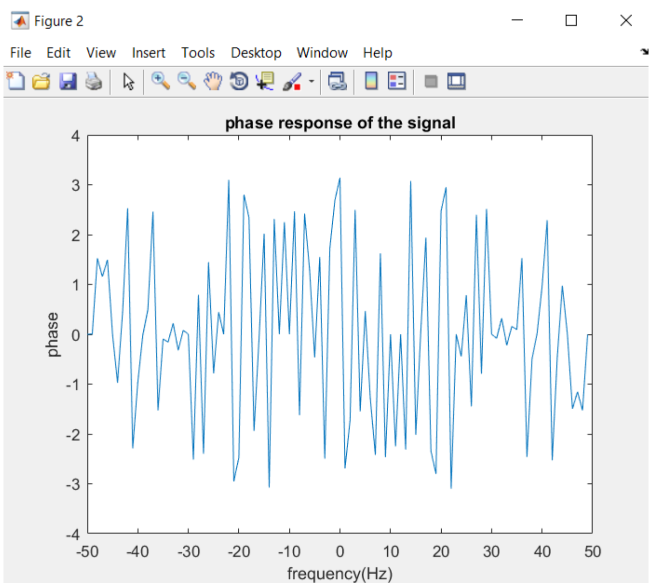
```
%q1
load 'signal.mat';
fs = 100;
%calculate fft and shift the plot
y = fft(x);
shifty = fftshift(y);
N = length(x);
w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2)-2*pi;
w = w/(2*pi)*fs;
%find the peak value and print
counter = 0;
for i = 1 : N
    if (shifty(i) > 10) && (w(i) > 0)
        counter = counter + 1;
        fprintf('tone %d is: %f Hz \n', counter, w(i));
    end
end
%plot
figure(1);
plot(w, abs(shifty));
xlabel('frequency(Hz)');
ylabel('magnitude');
title('magnitude response of the signal');
figure(2);
plot(w, angle(shifty));
xlabel('frequency(Hz)');
ylabel('phase');
title('phase response of the signal');
```

Explanation part:

Plot 1 below is the magnitude response of the signal:



Plot 2 below is the phase response of the signal:



There are 5 tones in the signal and their frequencies are: 10 Hz, 12 Hz, 23 Hz, 30 Hz and 45 Hz.

Question 2:

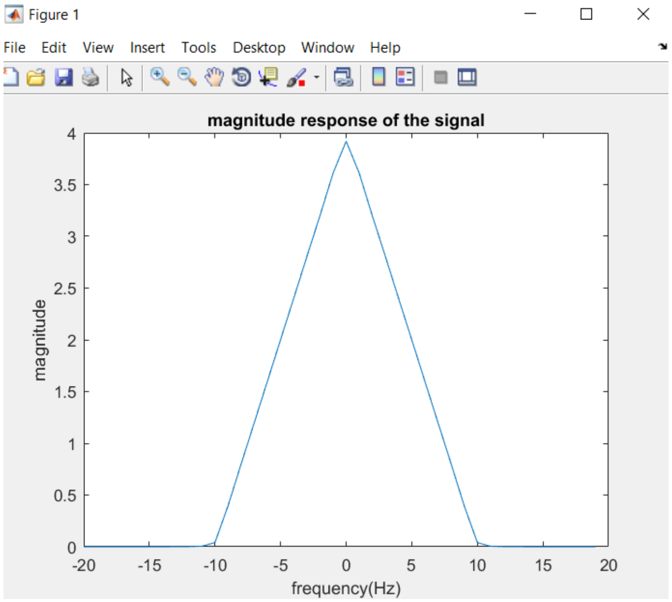
Code Part:

```
%q2
load 'samplerate.mat';
N = length(x);
w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2)-2*pi;
%1st part
fs = 40;
y1 = fft(x);
shifty1 = fftshift(y1);
t1 = 0: 1/fs : (N-1)/fs;
w = w/(2*pi)*fs;
figure(1);
plot(w, abs(shifty1));
xlabel('frequency(Hz)');
ylabel('magnitude');
title('magnitude response of the signal');
figure(2);
stem(t1, x);
xlabel('time(t)');
ylabel('magnitude');
title('time domain plots of the original signal');
%2nd part
x2 = upsample(x, 3);
N2 = length(x2);
y2 = fft(x2);
shifty2 = fftshift(y2);
t2 = 0: 1/(fs*3) : (N2-1)/(fs*3);
w2 = fftshift((0:N2-1)/N2*2*pi);
w2(1:N2/2) = w2(1:N2/2)-2*pi;
w2 = w2/(2*pi)*fs;
figure(3);
plot(w2, abs(shifty2));
xlabel('frequency(Hz)');
ylabel('magnitude');
title('magnitude response of the upsampling signal');
figure(4);
stem(t2, x2);
xlabel('time(t)');
ylabel('magnitude');
title('time domain plots of the upsampling signal');
%3rd part
%apply the ideal LPF
y3 = zeros(1, N2);
for i = 1 : N2
```

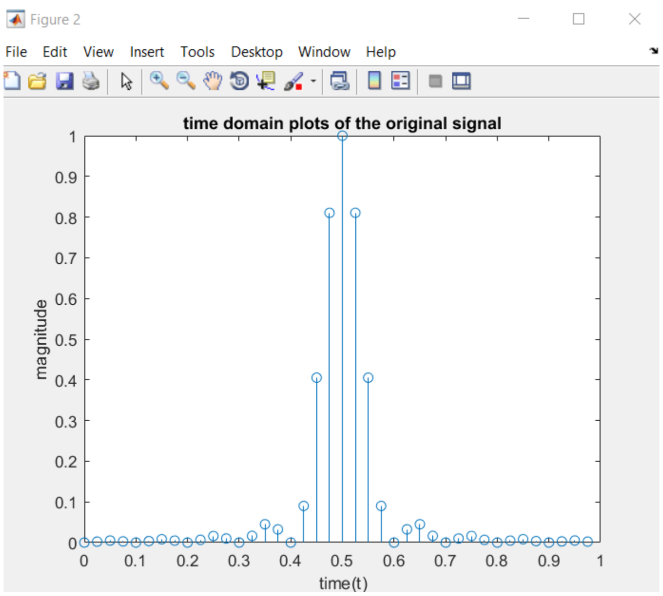
```
    if (w2(i) < -5) || (w2(i) > 5)
        y3(i) = 0;
    else
        y3(i) = shifty2(i);
    end
end
x3 = ifft(y3);
figure(5);
subplot(2, 1, 1);
plot(w2, abs(y3));
xlabel('frequency(Hz)');
ylabel('magnitude');
title('magnitude response of the upsampling signal after LPF');
subplot(2, 1, 2);
plot(t2, x3);
xlabel('time(t)');
ylabel('magnitude');
title('time domain plots of the upsampling signal after LPF');
%4th part
x4 = downsample(x3, 2);
N4 = length(x4);
y4 = fft(x4);
shifty4 = fftshift(y4);
t4 = 0: 1/(fs*3/2) : (N4-1)/(fs*3/2);
w4 = fftshift((0:N4-1)/N4*2*pi);
w4(1:N4/2) = w4(1:N4/2)-2*pi;
w4 = w4/(2*pi)*fs;

figure(6);
subplot(2, 1, 1);
plot(w4, abs(shifty4));
xlabel('frequency(Hz)');
ylabel('magnitude');
title('magnitude response of the downsampling signal after
LPF');
subplot(2, 1, 2);
plot(t4, x4);
xlabel('time(t)');
ylabel('magnitude');
title('time domain plots of the downsampling signal after LPF');
```

Explanation part:

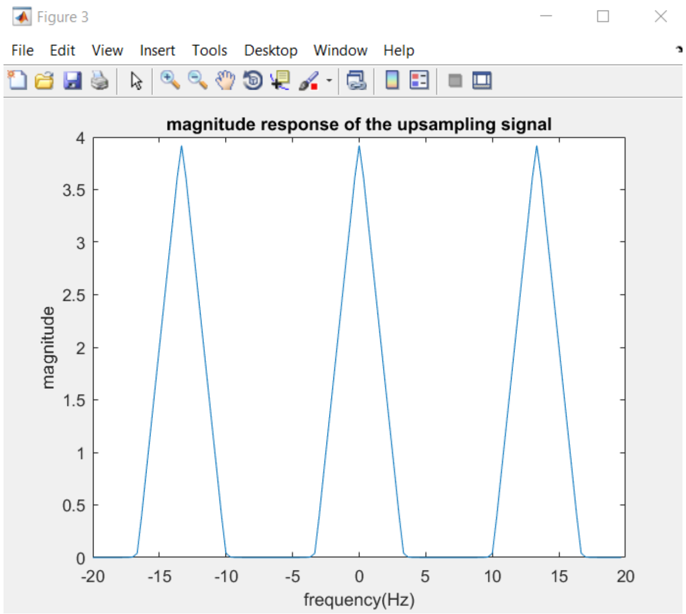
(1): plot 1 below is the magnitude spectrum



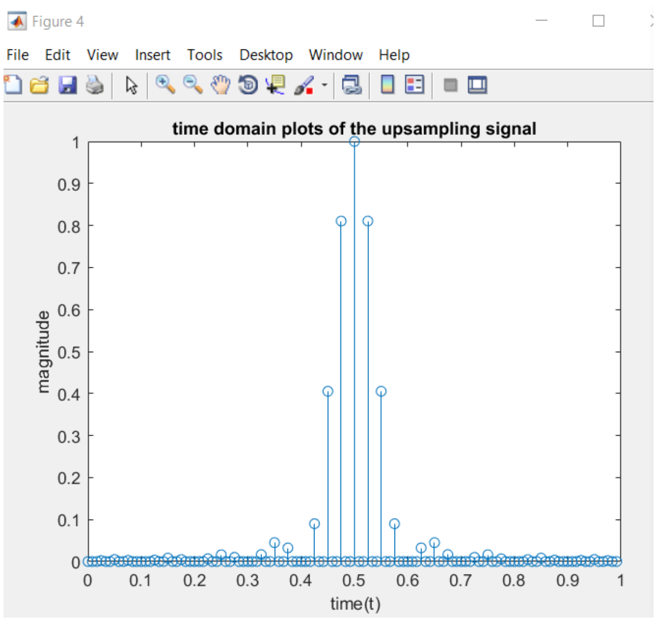
Plot 2 below is the time domain plot of the signal



(2): plot 3 below is the resulting magnitude spectrum



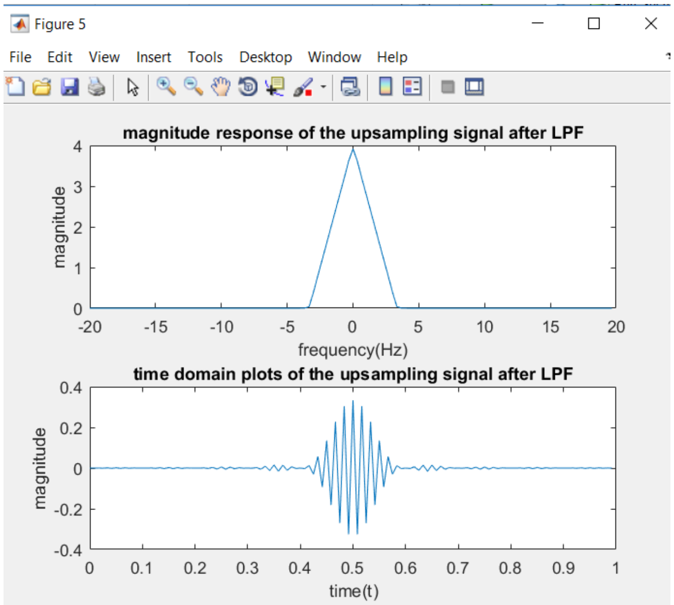
Plot 4 below is the time domain plots for the new signal:



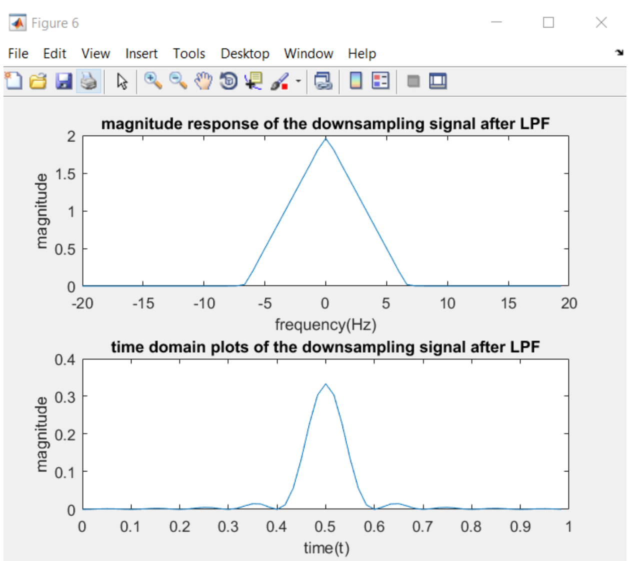
The cutoff frequency in part 2 is smaller than the one in part 1 due to the scaling property of the Fourier transform. For an up-sampler:

If x(t) leads to X(w), and output y(t) leads to Y(w), and y(t) is the up-sampled version of x(t), then Y(w) = X(L*w) suppose L is the upsample parameter.

(3): The plot of the magnitude spectrum and corresponding time-domain waveform after LPF is plot in plot 5 below:



(4): The plot of the magnitude spectrum and corresponding time-domain waveform after downsampling by 2 is plot in plot 6 below:



Final sample rate is 40*3/2 = 60 Hz.

The maximum value of D we can choose is that $20 = 10/3*D_{max}$

$D_{max} = 6$

Question 3:

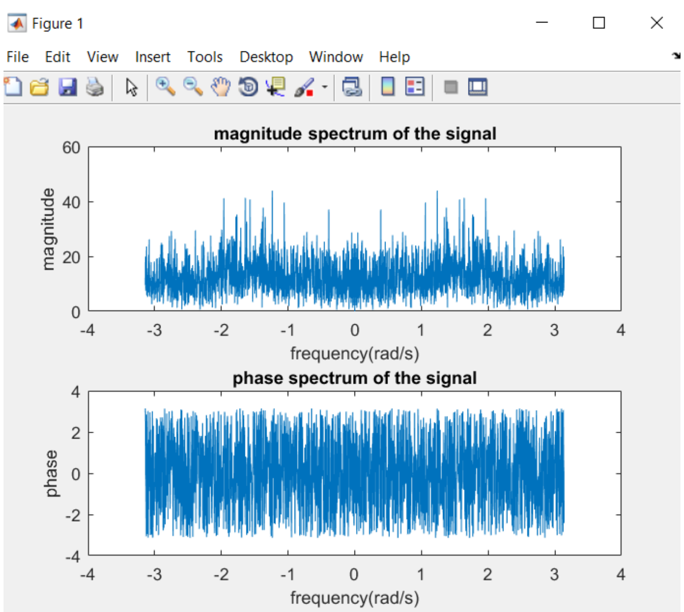
Code part:

```
%q3
load 'q1_signal.mat';
N = length(x);
y = fft(x);
%1st part
shifty = fftshift(y);
w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2)-2*pi;
figure(1);
subplot(2, 1, 1);
plot(w, abs(shifty));
xlabel('frequency(rad/s)');
ylabel('magnitude');
title('magnitude spectrum of the signal');
subplot(2, 1, 2);
plot(w, angle(shifty));
xlabel('frequency(rad/s)');
ylabel('phase');
title('phase spectrum of the signal');
%2nd part
ysig = zeros(1, N);
siglen = length(sig);
for i = 1 : N
    temp = mod(i, siglen);
    if temp == 0
        temp = siglen;
    end
    ysig(i) = sig(temp);
end
yysig = fft(ysig);
shiftysig = fftshift(yysig);


figure(2);
subplot(2, 1, 1);
plot(w, abs(shiftysig));
xlabel('frequency(rad/s)');
ylabel('magnitude');
title('magnitude spectrum of the sig');
subplot(2, 1, 2);
plot(w, angle(shiftysig));
xlabel('frequency(rad/s)');
ylabel('phase');
title('phase spectrum of the sig');
```

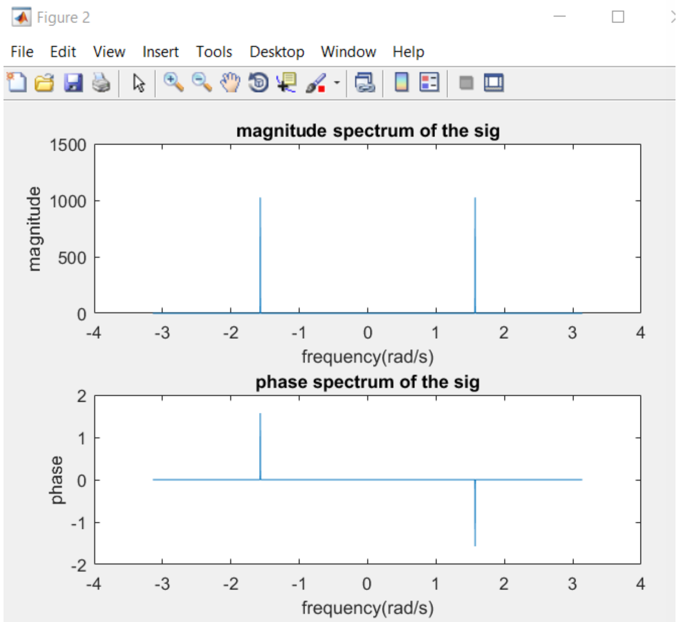
```
%3rd part
y3 = zeros(1, N);
for i = 1 : N
    if ((w(i) < -1.54) && (w(i) > -1.6)) || ((w(i) > 1.54) &&
(w(i) < 1.6))
        y3(i) = shifty(i);
    else
        y3(i) = 0;
    end
end
x3 = ifft(y3);
figure(3);
plot(x3);
xlabel('samples');
ylabel('magnitude');
title('bandpass filter the signal from 1.54 to 1.6');
```

Explanation part:

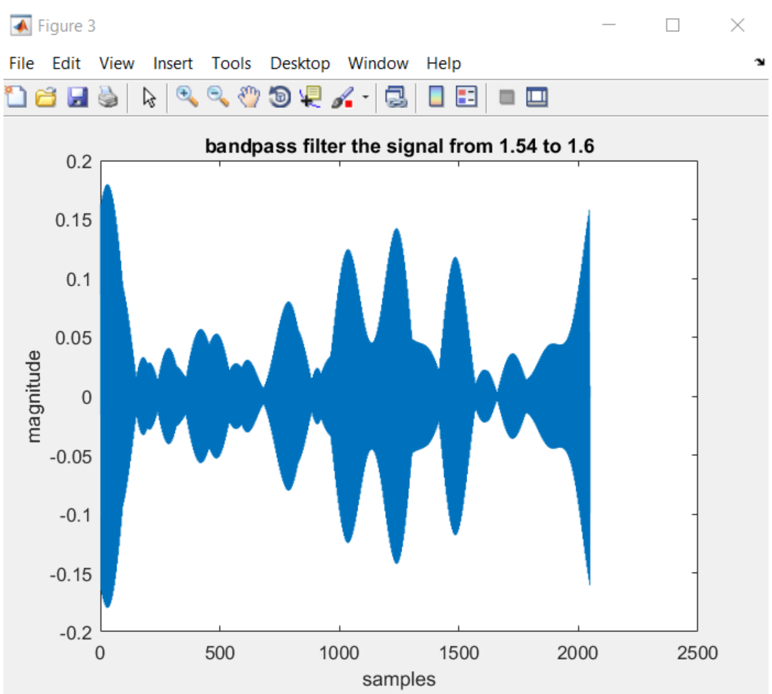
(1): the magnitude and phase spectrum of x is plot below:



(2): the magnitude and phase spectrum of sig is plot below:



(3): Since the peak value of the magnitude response of sig is at 1.57 rad/s, I apply a bandpass filter between 1.54 and 1.6 rad/s and the resulting signal in time domain is plot below:



There are at least 6 times ‘sig’ occurred in ‘x’ since there are 6 main peaks on the plot.

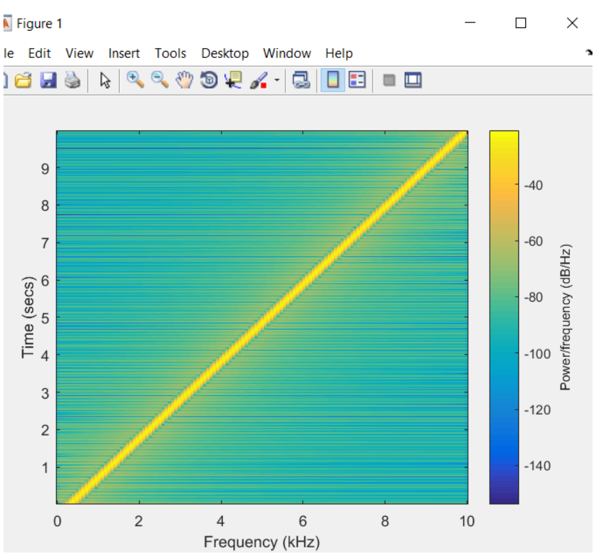
Question 4:

Code part:

```
%q4
load 'q2_signal.mat';
%soundsc(x, fs);
figure(1);
spectrogram(x, 256, 128, 256, fs);

%part2
N = round(length(x)/2);

x2 = zeros(1, N);
for i = 1 : N
    x2(i) = x(2*i-1);
end
figure(2);
%soundsc(x2, fs/2);
spectrogram(x2, 256, 128, 256, fs/2);

%part3:
y = resample(x, 1, 2);
figure(3);
%soundsc(y, fs/2);
spectrogram(y, 256, 128, 256, fs/2);
```

Explanation part:

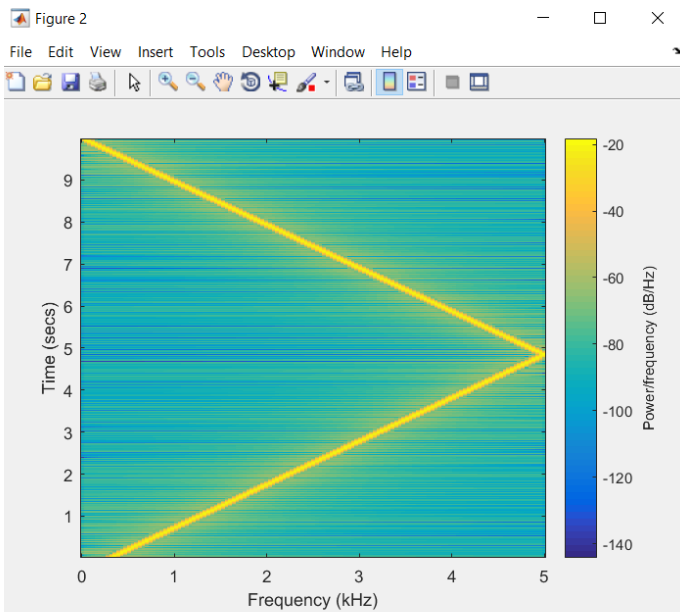
(1): The spectrogram of the signal x is plot below:



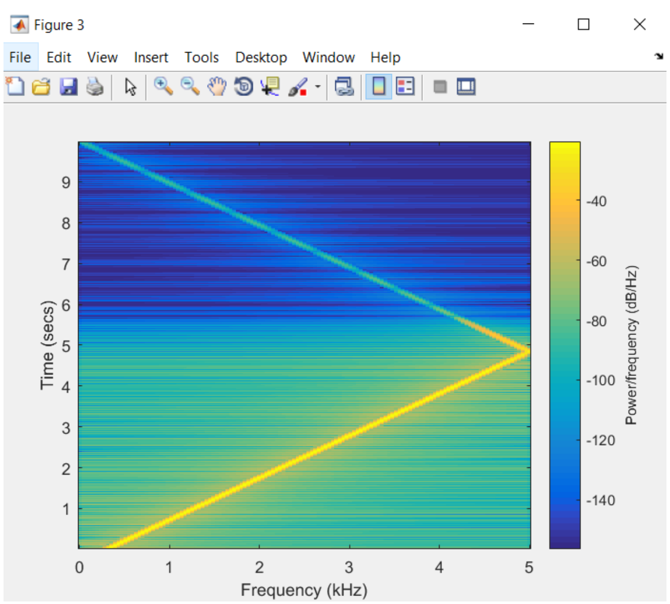
The sound hears like a continuously increasing pitch.

(2): After desampling by removing every even element of ‘x’, it will cause the aliasing between each replica in the frequency spectrum. The pitch will go up and come down instead of increasing continuously.

The spectrogram of the new signal is plotting below:



(3): The correct downsample ‘x’ has spectrogram like the one plot below:



The resample function has the lowpass filter built in which can be used to prevent the aliasing during resampling. And it hears like the pitch continuously goes up which is the same as the one in the original signal.