Problem 1. Matlab code and figures for bandpass sampling problem are shown below.

```matlab
function hmwk5p1()
    % Bandpass sampling problem (This is actually Hmwk 5)
    % x(t) = 100 exp(-200|t|) sin(2 pi 10^5 t) 
    % exp(-200|t|) --> 400/(200^2 + (2 pi f)^2)
    figure(1)
    f=[-2000:5:2000];
    A=400*1600./(800^2 + (2*pi*f)^2);
    plot(f,20*log10(A))
    title('V(f)');xlabel('Frequency, f, Hz'); ylabel('V(f)')
    Fs=1e6;
    tR=0.01;dt=1/Fs;
    t=[-tR:dt:tR];
    x=400*exp(-800*abs(t)).*sin(2*pi*104000*t);
    figure(2)
    plot(t,x)
    title('v(t)');xlabel('Time, t, sec'); ylabel('v(t)')
    X=fft(x);
    f=[0:length(X)-1]*Fs/length(X);
    Xa=(3.2e5/j)./(800^2 + (2*pi*(f-104000)).^2); % Analog transform of BP signal
    figure(3)
    plot(f,20*log10(abs(Xa))),'f,20*log10(Fs*abs(Xa))',
    title('X_a(f) *F_s and DFT Magnitude, in dB');xlabel('Frequency, f, Hz');
    ylabel('|X_a(f)|/times F_s, |X(k)|')
    % Bandpass filter to [100KHz, 110KHz] range first
    % Then K=floor(f1/2W)=floor(100,000/(2*10,000))=5
    % So. (1/T) = f1/K = 100,000/5 = 20 kHz.
    % Since original data at 1 MHz, just subsample by factor of 50.
    xbp=x(1:50:end);
    Xbp=fft(xbp,max(length(xbp),1024));
    fbp=[0:length(Xbp)-1]*20000/length(Xbp);
    figure(4)
    plot(fbp+100000,20*log10(abs(Xbp))) % 5*F_bp = 100,000 Hz added to f in plot
    title('Bandpass Sampling Spectrum');xlabel('Frequency, f, Hz');
    ylabel('|X(k)|')
    % Repeat with iid noise
    sig=20; % added noise standard deviation
    xn=400*exp(-800*abs(t)).*sin(2*pi*104000*t)+sig*randn(1,length(t));
    figure(5)
    plot(t,xn)
    title('v(t) With IID Gaussian Noise, \sigma = 20');xlabel('Time, t, sec');
    ylabel('v(t) + noise')
    Xn=fft(xn);
    f=[0:length(Xn)-1]*Fs/length(Xn);
    figure(6)
    plot(f,20*log10(abs(Xn))
    title('DFT Magnitude for x_a(t)+noise, in dB, 1 MHz sampling rate');xlabel('Frequency, f, Hz');
    ylabel('|X(k)|, dB')
    % Bandpass filter to [100KHz, 110KHz] range first
    % Then K=floor(f1/2W)=floor(100,000/(2*10,000))=5
    % So. (1/T) = f1/K = 100,000/5 = 20 kHz.
    % Since original data at 1 MHz, just subsample by factor of 50.
    xbpn=xn(1:50:end);
    Xbpn=fft(xbpn,max(length(xbpn),1024));
    fbp=[0:length(Xbpn)-1]*20000/length(Xbpn);
    figure(7)
```
plot(fb+100000, 20*log10(abs(Xb)))
title('Bandpass Sampling Spectrum, Without Bandpass Filter'); xlabel('Frequency, f, Hz'), ylabel('|X(k)|')
legend('Bandpass Frequency Plus 100 kHz')

% Repeat with bandpass filter [100, 110] kHz
h=fir1(100, [100 110]/500, 'bandpass');
y=conv(xn, h);
yb=fft(yb, max(length(yb), 1024));
fbp=[0:length(Yb)-1]*20000/length(Yb);
figure(8)
plot(fb+100000, 20*log10(abs(Yb)));
title('Bandpass Sampling Spectrum With Bandpass Filtering First'); xlabel('Frequency, f, Hz'), ylabel('|X(k)|')
legend('Bandpass Frequency Plus 100 kHz')
end

a) $V(f) = \frac{64000}{64000 + (2\pi f)^2}$. Bandwidth (at -30 dB level) is about 710 Hz

![Graph of V(f)](image)

b) $X_a(f) = \frac{1}{2j}[V(f - f_0) - V(f + f_0)]$, $f_0 = 104,000$ Hz. From part a), the bandwidth is $2 \times 710 = 1,420$ Hz. From the figure below, it is measured (in the Matlab plot at about 1,400 Hz. $x_a(t)$ is shown below (the sinusoid oscillates so far relative to the scale of the plot that the individual oscillations in the signal are not apparent).
$X_a(f)$ and the DFT magnitude. The analog spectrum is scaled by the sampling rate. Negligible aliasing is apparent, and negligible effect of signal truncation (since the time signal has negligible energy before time $-0.01$ sec and after time $0.01$ sec).
c) Bandpass sampling. $f_1 = 100$ kHz, $f_2 = 110$ kHz, so $K = \left| \frac{f_1}{2 \times (f_2 - f_1)} \right| = 5$.

Thus $\left( \frac{1}{T} \right)_{BP} = \frac{f}{K} = 20,000$ Hz. Applying bandpass sampling (subsampling the 1 MHz sampled data by a factor of 50) and then computing the DFT yields the spectrum below (with 100,000 Hz added to the horizontal axis values). This is the 5th translate.

![Bandpass Sampling Spectrum](image)

f) Additive white Gaussian noise samples are added to the original time samples, resulting in the signal shown below.

![v(t) With IID Gaussian Noise, σ = 20](image)
f) Direct bandpass sampling without first applying a bandpass filter. The first figure below shows the direct DFT of the noise signal (with sampling rate 1 MHz). Note that the noise floor is 35 to 40 dB less than the peaks in the spectrum.

Bandpass sampling without applying a bandpass filter yields the spectrum below. Note that the noise floor, relative to the peak in the spectrum, is significantly higher than in the figure above. It is unreasonable to try to measure bandwidth at the -30 dB from peak level. Without pre-filtering using a bandpass filter, out-of-band noise is aliased into the signal frequency band, resulting in the larger amount of noise in the spectrum.
f, ii) First apply a bandpass filter, then bandpass sample. In this case the noise floor is reduced, and a bandwidth measurement at the -30 dB from peak level is reasonable and the resulting value is consistent with the earlier result.

Problem 3

```matlab
>> f=[0:1:2400];
>> G=(1-exp(-j*2*pi*f/2400))/(1-0.9*exp(-j*2*pi*f/2400));
>> figure(1)
>> figure(2)
>> plot(f,abs(G))
>> H=(1-exp(-j*2*pi*f/2400))/(1-0.9*exp(-j*2*pi*f/2400));
>> G=(1-exp(-j*2*pi*f*20/2400))/(1-0.9*exp(-j*2*pi*f*20/2400));
>> subplot(2,1,1)
>> plot(f,abs(H))
>> title('Prototype Filter Frequency Response, H(z) = (1-z^-1)/(1-\alpha z^-1)')
>> xlabel('Frequency, f, Hz')
>> ylabel('|H(e^{j2\pi f})|')
>> subplot(2,1,2)
>> plot(f,abs(G))
>> title('Comb Filter Frequency Response, G(z) = (1-z^-20)/(1-\alpha z^{-20})')
>> ylabel('|G(e^{j2\pi f})|')
>> xlabel('Frequency, f, Hz')
```
\[ x[n] = \begin{pmatrix} 1 & 1 & -1 & -2 & 1 \end{pmatrix}, \quad h[n] = \begin{pmatrix} 2 & 2 & 1 \end{pmatrix} \]

\[ n_x = 5 \quad n_h = 3 \]

a) \( N = 6 \) \[ x[0] \oplus [h \circ 0] = y[n] = [1, 5, 1, -4, -2, -1] \]

b) \( N = 10 \) \[ x[0] \oplus [h \circ 0] = y[n] = [2, 4, 1, -4, -2, -1, -3, 3, 0, 0] \]

c) Linear convolution: \[ y[n] = x[n] * h[n] = [2, 4, 1, -4, -2, -1, -3, 3, 0, 0] \]

Smallest \( N = 5+4-1 = 8 \) for circular convolution to be equivalent to linear convolution.

Note: \[ y[n] = \sum_{r=0}^{\infty} x[n-r]h[r] \]

3) Use prototype filter with \( f = \frac{1}{2\pi} \text{Hz} \), \( H(2) = \frac{1 - 2}{1 - 0.3^2} \). Then \( H(2) = G(\xi) \) has zeros at \( 1 - \frac{1}{\xi} = 0 \Rightarrow \xi = \xi_k \), \( k = 0, 1, 2, ... \)

\[ G(\xi) = H(2) = \frac{1 - \xi}{1 - 0.3^2} \text{ for } \xi = \xi_k \]

\[ \Rightarrow \xi = \frac{120}{2\pi} \text{ Hz} \]

\[ \frac{1}{2\pi} = \frac{120}{2\pi} \Rightarrow L = 20 \]

\[ \alpha = 0.9 \] in Matlab results in 3 dB point of notch at \(-3 \text{ dB point}\).
Comb filter frequency response, $\alpha = 0.9$. Spectral nulls at 0, 120, 240, ..., 1200 Hz.
EE 464 Homework 5 Solutions

4) \( x[n] = \begin{cases} \frac{1}{2} x(n), & n \text{ even} \\ 0, & n \text{ odd} \end{cases} \)
\( X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n} = \sum_{n=-\infty}^{\infty} \frac{1}{2} x(n) e^{-j\omega n} = \frac{1}{2} \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n} = X(e^{j\omega/2}) \)

b) \( x[n] = \frac{1}{2} x[2n] \)
\( X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n} = \sum_{n=-\infty}^{\infty} \frac{1}{2} x[2n] e^{-j\omega n} = \frac{1}{2} \sum_{n=-\infty}^{\infty} x[2n] e^{-j\omega n} = \frac{1}{2} X(e^{j\omega}) \)

Use \( X(e^{j\omega}) \) from part a) to find \( X(e^{j\omega}) \) for \( x[n] = \frac{1}{2} x[2n] \)

\( X(e^{j\omega}) \) for \( x[n] = \frac{1}{2} x[2n] \) is zero-padded in the frequency domain.

5) \( x[n] = \begin{cases} \frac{1}{2} x[2n], & k = 0, 1, 2, \ldots, 2N-1 \\ 0, & \text{otherwise} \end{cases} \)

a) \( X[k] = \begin{cases} x[0], & k = 0 \\ \frac{1}{2} x[2], & k = 1, 2, \ldots, 2N-1 \\ 0, & \text{otherwise} \end{cases} \)

b) \( X[k] = \begin{cases} x[0], & k = 0 \\ \frac{1}{2} x[2], & k = 1, 2, \ldots, 2N-1 \\ 0, & \text{otherwise} \end{cases} \)

This is zero-padded in the frequency domain.

6) \( x[n] = \begin{cases} \frac{1}{2} x[2n], & n = 0, 1, 2, \ldots, N-1 \\ 0, & \text{otherwise} \end{cases} \)

a) \( X[k] = \begin{cases} x[0], & k = 0 \\ \frac{1}{2} x[2], & k = 1, 2, \ldots, N-1 \\ 0, & \text{otherwise} \end{cases} \)

b) \( X[k] = \begin{cases} x[0], & k = 0 \\ \frac{1}{2} x[2], & k = 1, 2, \ldots, N-1 \\ 0, & \text{otherwise} \end{cases} \)

f) \( x[n] = \begin{cases} x[n], & n = 0, \ldots, N-1 \\ 0, & \text{otherwise} \end{cases} \)

\( X[k] = \begin{cases} x[k], & k = 0, \ldots, N-1 \\ 0, & \text{otherwise} \end{cases} \)
\[ \text{Figure C} \]
\[ \text{Figure D} \]
\[ \text{Figure E} \]

(i) \( y(t) = (y_0, y_1, y_2, \ldots, y_n) \) - periodic \( \Rightarrow \) \( y(t) \) is zero, non-trivial, \( \text{Figure (E)} \).

(ii) \( y(t) = (y_0, y_1, y_2, \ldots) \) - zero in between \( \Rightarrow \) \( y(t) \) is periodic extension, \( y(t) = (y_0, y_1, y_2, \ldots) \) - \( \text{Figure (E)} \).

(iii) \( y(t) = (y_0, y_1, y_2, \ldots) \) - zero padding
\[ y(t) = \begin{cases} X_k & \text{even} \\ 0 & \text{odd} \end{cases} \] - \( \text{Figure (E)} \) for \( y(t) \).