Expt3

```
% Load ECG signal
data = load('100m.mat');
fs = 1000; % Sampling frequency (adjust if needed)

% Check field names in the loaded data structure
fieldnames(data)

% Extract ECG signal (replace 'val' with the correct field name if different)
ecg_signal = data.val;

% Ensure the signal is a row vector for processing
% ecg_signal = ecg_signal(:);

% Design and apply a high-pass filter
hp_cutoff = 0.5; % High-pass cutoff frequency in Hz
hp_order = 4; % Filter order
[b_hp, a_hp] = butter(hp_order, hp_cutoff / (fs / 2), 'high');
ecg_highpass = filtfilt(b_hp, a_hp, ecg_signal);

% Design and apply a low-pass filter
lp_cutoff = 40; % Low-pass cutoff frequency in Hz
lp_order = 4; % Filter order
[b_lp, a_lp] = butter(lp_order, lp_cutoff / (fs / 2), 'low');
ecg_lowpass = filtfilt(b_lp, a_lp, ecg_highpass);

% Time vector for plotting
time = (0:length(ecg_signal) - 1) / fs;

% Plot original and processed ECG signals
figure;

% Original ECG signal
subplot(3, 1, 1);
plot(time, ecg_signal);
title('Original ECG Signal');
xlabel('Time (s)');
ylabel('Amplitude');

% High-pass filtered ECG signal
subplot(3, 1, 2);
plot(time, ecg_highpass);
title('ECG Signal After High-Pass Filtering');
xlabel('Time (s)');
ylabel('Amplitude');

% Low-pass filtered ECG signal
```

```
subplot(3, 1, 3);
plot(time, ecg_lowpass);
title('ECG Signal After Low-Pass Filtering');
xlabel('Time (s)');
ylabel('Amplitude');


EXpt4
clear;

%load ecg signal
fid = fopen('rec_1.dat','r');
data = fread(fid, 'int16');
fclose(fid);
ecg_signal = data(1:2:end);

% define parameters
fs = 500;
t = (0: length(ecg_signal)-1)/fs;

%high pass filtering
h_cutoff = 0.5;
[b_h, a_h] = butter(2, h_cutoff/(fs/2), 'high');
h_filtered = filtfilt(b_h, a_h, ecg_signal);

%low pass filtering
l_cutoff = 50;
[b_l, a_l] = butter(2, l_cutoff/(fs/2), 'low');
l_filtered = filtfilt(b_l, a_l, h_filtered);

%detect peaks
[peaks, locs] = findpeaks(l_filtered, 'MinPeakHeight', 0.5, 'MinPeakDistance', fs/2);


%plot each fckin thing
figure;
subplot(4,1,1);
plot(t, ecg_signal);
title("Original signal");
subplot(4,1,2);
plot(t, h_filtered);
title("High pass filtered signal");
subplot(4,1,3);
plot(t, l_filtered);
title("Low pass filtered signal");
subplot(4,1,4);
plot(t, l_filtered);
hold on;
```

```
plot(t(locs), peaks, 'ro');
title("Ecg signal with peaks");

% calculate heart rate
rr_interval = diff(locs)/fs;
heart_rate = 60 / mean(rr_interval);
disp("Heart rate is: ");
disp(heart_rate);
```

EXpt6

```
clear;
% read the ecg signal
fid = fopen('rec_1.dat', 'r');
data = fread(fid, 'int16');
fclose(fid);
ecg_signal = data(1:2:end);

%define parameters
fs = 250;
t = (0: length(ecg_signal) - 1)/fs;
N = length(ecg_signal);
ecg_fft = fft(ecg_signal);

figure;
subplot(2,1,1);
plot(t,ecg_signal);
title("Original Signal");
subplot(2,1,2);
plot((1:floor(N/2)), abs(ecg_fft(1:floor(N/2))));
title("FFT of Signal");
```

expt7

```
clear;

%read ecg
% fid = fopen('rec_1.dat','r');
% data = fread(fid, 'int16');
% fclose(fid);
% ecg_signal = data(1:2:end);
data = load('100m.mat');
ecg_signal = data.val;

%define parameters and awgn
fs = 500;
t = (0:length(ecg_signal)-1)/fs;
noise_var = 0.5;  % You can adjust this value
```

```
noisy_signal = awgn(ecg_signal, 10*log10(1/noise_var), 'measured');


%cutoff frequencies
l_cutoff = 30;
h_cutoff = 0.5;
b_cutoff = [30, 249];

%low pass
[b_l, a_l] = butter(2, l_cutoff/(fs/2), 'low');
l_filtered = filtfilt(b_l, a_l, noisy_signal);

%high pass
[b_h, a_h] = butter(2, h_cutoff/(fs/2), 'high');
h_filtered = filtfilt(b_h, a_h, noisy_signal);

%notch filter (band stop)
[b_n, a_n] = butter(2, b_cutoff/(fs/2), 'stop');
n_filtered = filtfilt(b_n, a_n, noisy_signal);

%plot all figures
figure;
subplot(5,1,1);
plot(t, ecg_signal);
title('Original Signal'); xlabel('Time (s)'); ylabel('Amplitude');
subplot(5,1,2);
plot(t, noisy_signal);
title('Noisy Signal'); xlabel('Time (s)'); ylabel('Amplitude');
subplot(5,1,3);
plot(t, l_filtered);
title('Low Pass filtered Signal'); xlabel('Time (s)'); ylabel('Amplitude');
subplot(5,1,4);
plot(t, h_filtered);
title('High Pass filtered Signal'); xlabel('Time (s)'); ylabel('Amplitude');
subplot(5,1,5);
plot(t, n_filtered);
title('Notch Filtered Signal'); xlabel('Time (s)'); ylabel('Amplitude');


exp5
% Load the EEG data
data = load('subject00_1_edfm.mat');

% Extract fieldnames and EEG signal
fields = fieldnames(data);
disp(fields); % Display the field names to confirm structure
eeg_sig = data.eeg_signal; % Assuming the signal is stored under the 'val' field
num_ch = size(eeg_sig, 1); % Number of channels
```

```
% Determine the grid size for subplots
rows = ceil(num_ch / 2); % Dynamically set rows to fit all channels
col = 2; % Fixed number of columns

% Plot each EEG channel
figure; % Create a new figure
for channel = 1:num_ch
   subplot(rows, col, channel); % Dynamically adjust subplot indices
   plot(eeg_sig(channel, :), 'LineWidth', 1); % Plot the EEG signal for each channel
   title(['EEG Channel ', num2str(channel)], 'FontSize', 10);
   grid on;
   xlabel('Sample Points');
   ylabel('Amplitude');
end
```