```matlab
%--------------------------------------------------------------------%
%---------------Load the male vowel audio file-----------------------%
[male_vowel, fs] = audioread('hood_m.wav');

%Play the male vowel audio file
sound(male_vowel, fs);

% Load the female vowel audio file
[female_vowel, fs] = audioread('hood_f.wav');

%Play the female vowel audio file
sound(female_vowel, fs);

%--------------------------------------------------------------------%
%----------------Plotting the audio signals--------------------------%
%--------------Plot the male vowel audio signal----------------------%
time = (0:length(male_vowel) - 1) ./ fs;
figure(1);
subplot(2, 2, 1);
plot(time, male_vowel, 'b');
xlabel('Time(s)');
ylabel('Amplitude');
title('Speech Audio Signal(Male Vowel)');
grid on;

%--------------------------------------------------------------------%
%--------------Plot the male vowel audio signal-----------------------%
time = (0:length(female_vowel) - 1) ./ fs;
figure(1);
subplot(2, 2, 3);
plot(time, female_vowel, 'b');
xlabel('Time(s)');
ylabel('Amplitude');
title('Speech Audio Signal(Female vowel)');
grid on;

%--------------------------------------------------------------------%
%------------Plot spectrogram for male and female vowels-------------%
figure(1);
subplot(2, 2, 2);
spectrogram(male_vowel,100,80,100,fs,'yaxis')
title('Male Vowel Signal Spectrogram')
figure(1);
subplot(2, 2, 4);
spectrogram(female_vowel,100,80,100,fs,'yaxis')
title('Female Vowel Signal Spectrogram')

%--------------------------------------------------------------------%
%-----------------Define the analysis parameters---------------------%
segment_length = 0.1;  % change segment length for analysis
frame_length = round(segment_length .* fs);
%------------Extract segments for male and female vowels-------------%
male_segment = male_vowel(1:frame_length);
female_segment = female_vowel(1:frame_length);

%--------------------------------------------------------------------%
%---------------------Plot the male segment -------------------------%
time = (0:length(male_segment) - 1) ./ fs;
figure(2);
```

```
subplot(2, 2, 1);
plot(time, male_segment, 'b');
xlabel('Time (s)');
ylabel('Amplitude');
title('Speech Audio Signal(Male segment)');
grid on;

%----------------------------------------------------------------------%
%---------------------Plot the Female vowl segment --------------------%
time = (0:length(female_segment) - 1) ./ fs;
figure(2);
subplot(2, 2, 3);
plot(time, female_segment, 'b');
xlabel('Time (s)');
ylabel('Amplitude');
title('Speech Audio Signal(Female segment)');
grid on;

%-----------------------------------------------------------------------%
%----------Plot spectrogram for male and female segmented signals-------%
figure(2);
subplot(2, 2, 2);
spectrogram(male_segment,100,80,100,fs,'yaxis')
title('Male Segment Signal Spectrogram')
figure(2);
subplot(2, 2, 4);
spectrogram(female_segment,100,80,100,fs,'yaxis')
title('Female Segment Signal Spectrogram')

%--------------------------------------------------------------------%
%-----------------------LPC model order------------------------------%
lpc_order = 40;

%-------Estimate LPC coefficients for male and female vowels----------%
male_lpc_coeffs = lpc(male_segment, lpc_order);
female_lpc_coeffs = lpc(female_segment, lpc_order);

%--------------------------------------------------------------------%
%---------Plot the LPC filter response and amplitude spectrum--------%
figure(3);
subplot(2, 1, 1);
freqz(1, male_lpc_coeffs);  % Frequency response of male LPC filter
title('LPC Filter Response (Male Vowel)');

figure(4);
subplot(2, 1, 1);
amplitude_spectrum_male = 20 .* log10(abs(fft(male_segment)));
plot(linspace(0, fs, length(amplitude_spectrum_male)), amplitude_spectrum_male);
title('Amplitude Spectrum (Male Vowel)');
xlabel('Frequency (Hz)');
ylabel('Amplitude (dB)');

figure(5);
subplot(2, 1, 1);
freqz(1, female_lpc_coeffs);  % Frequency response of female LPC filter
title('LPC Filter Response (Female Vowel)');

figure(4);
subplot(2, 1, 2);
```

```
amplitude_spectrum_female = 20 .* log10(abs(fft(female_segment)));
plot(linspace(0, fs, length(amplitude_spectrum_female)),
amplitude_spectrum_female);
title('Amplitude Spectrum (Female Vowel)');
xlabel('Frequency (Hz)');
ylabel('Amplitude (dB)');

%--------------------------------------------------------------------%
%-------Estimate the first three formant frequencies for male vowel---%
[formants_male_vowel, bandwidths_male_vowel] = formants(male_lpc_coeffs, fs);
fprintf('Male Vowel Formant Frequencies (Hz):\n');
for i = 1:3
    fprintf('Formant %d: %.2f Hz\n', i, formants_male_vowel(i));
end

%-----Plotting the first three formant frequencies for male vowel-----%
figure(4);
subplot(2, 1, 1);
hold on;
for i = 1:3
 %-Plot vertical lines to represent formant frequencies for male vowel-%
    line([formants_male_vowel(i), formants_male_vowel(i)], ylim, 'Color', 'r',
'LineStyle', '--', 'DisplayName', sprintf('Formant %d', i));
end
legend;
hold off;

%---------------------------------------------------------------------%
%-----Estimate the first three formant frequencies for female vowel---%
[formants_female_vowel, bandwidths_female_vowel] = formants(female_lpc_coeffs,
fs);
fprintf('Female Vowel Formant Frequencies (Hz):\n');
for i = 1:3
    fprintf('Formant %d: %.2f Hz\n', i, formants_female_vowel(i));
end

%-----Plotting the first three formant frequencies for female vowel----%
figure(4);
subplot(2, 1, 2);
hold on;
for i = 1:3
 %-Plot vertical lines to represent formant frequencies for female vowel-%
    line([formants_female_vowel(i), formants_female_vowel(i)], ylim, 'Color', 'r',
'LineStyle', '--', 'DisplayName', sprintf('Formant %d', i));
end
legend;
hold off;
%--------------------------------------------------------------------%
%---Estimate fundamental frequencies of male and female vowels-------%
[f0_male, voicing_male] = pitch(male_segment, fs);
mean_f0_male = mean(f0_male);
[f0_female, voicing_female] = pitch(female_segment, fs);
mean_f0_female = mean(f0_female);
%--------------------------------------------------------------------%
%----------------------Display the results---------------------------%
disp(['Estimated F0 for male (mean): ' num2str(mean_f0_male) ' Hz']);
disp(['Estimated F0 for female (mean): ' num2str(mean_f0_female) ' Hz']);

%--------------------------------------------------------------------%
```

```
%-------Duration of the impulse train (roughly 1 second)-------------%
duration = 1;

%--------Calculate the number of impulses to generate the train---------%
number_of_impulses = round(duration .* fs);

%--------------Generate the impulse train for male and female vowels-----%
impulse_train_period_male = round(fs ./mean_f0_male);
impulse_train_period_female = round(fs ./mean_f0_female);

impulse_train_for_male = min(number_of_impulses,impulse_train_period_male);
impulse_train_male_vowel = zeros(1,number_of_impulses);
impulse_train_male_vowel(1:impulse_train_for_male:end) = 1;

impulse_train_for_female = min(number_of_impulses,impulse_train_period_female);
impulse_train_female_vowel = zeros(1,number_of_impulses);
impulse_train_female_vowel(1:impulse_train_for_female:end) = 1;

synthesized_sound_male_vowel =filter(1, male_lpc_coeffs ,
impulse_train_male_vowel);
synthesized_sound_male_vowel = synthesized_sound_male_vowel ./
max(abs(synthesized_sound_male_vowel));
synthesized_sound_female_vowel =filter(1, female_lpc_coeffs ,
impulse_train_female_vowel);
synthesized_sound_female_vowel = synthesized_sound_female_vowel ./
max(abs(synthesized_sound_female_vowel));

%------------------------------------------------------------------%
%-------------Plot the impulse train for male vowel------------------%
figure(15);
subplot(2,1,1);
time = (0:number_of_impulses-1) ./ fs;
stem(time, impulse_train_male_vowel, 'r', 'filled');
xlabel('Time (s)');
ylabel('Amplitude');
title('Periodic Impulse Train for Male');

%-------------Plot the impulse train for female vowel-----------------%
%time = (0:number_of_impulses-1) ./ fs;
figure(15);
subplot(2,1,2);
stem(time, impulse_train_female_vowel, 'r', 'filled');
xlabel('Time(s)');
ylabel('Amplitude');
title('Periodic Impulse Train for Female');

%------------------------------------------------------------------%
%-----------Plot the synthesized speech male vowel-------------------%
time = (0:length(synthesized_sound_male_vowel) - 1) ./ fs;
figure(17);
subplot(2,1,1);
plot(time, synthesized_sound_male_vowel);
xlabel('Time (s)');
ylabel('Amplitude');
title('Synthesized Speech Male vowel');

% Plot the synthesized speech female
time = (0:length(synthesized_sound_female_vowel) - 1) ./ fs;
figure(17);
```

```
subplot(2,1,2);
plot(time, synthesized_sound_female_vowel);
xlabel('Time(s)');
ylabel('Amplitude');
title('Synthesized Speech Female vowel');

%--------------------------------------------------------------------%
%-----------------Play the synthesized male vowel speech-------------%
soundsc(synthesized_sound_male_vowel, fs);

%-----------------Play the synthesized female vowel speech-----------%
soundsc(synthesized_sound_female_vowel, fs);

%--------------------------------------------------------------------%
%--------------Save the synthesized male vowel sound-----------------%
file_name_male = 'Manish_synthesized_output_male_vowel.wav';
audiowrite(file_name_male, synthesized_sound_male_vowel, fs);

%--------------Save the synthesized female vowel sound---------------%
file_name_female = 'Manish_synthesized_output_female_vowel.wav';
audiowrite(file_name_female, synthesized_sound_female_vowel, fs);

%--------------------------------------------------------------------%
%------------------Start of functions--------------------------------%
%----------Function to find formant frequencies----------------------%
function [formants, bandwidths] = formants(coeffs, fs)
    rts = roots(coeffs);
    rts = rts(imag(rts) >= 0);
    angz = atan2(imag(rts), real(rts));
    [~, indices] = sort(angz);

%----------Calculate formants-------------%
    formants = sort(angz) .* (fs ./ (2 .* pi));

%---------Calculate bandwidths------------%
    bandwidths = -1 ./ 2 .* (fs / (2 .* pi)) .* log(abs(rts(indices)));

%----------Loop for error handling---------%
%This loop checks and handle formants with identical frequencies and zeros
    for i = 1:length(formants)
        if formants(i) == 0
%This loop searches for the next neighboring non-zero formants
            for j = i+1:length(formants)
                if formants(j) > 0
                    formants(i) = formants(j);
                    break;
                end
            end
        elseif i > 1 && formants(i) == formants(i - 1)
            %Loop to search the next neighboring non-identical formants
            j = i + 1;
            while j <= length(formants) && formants(j) == formants(i)
                j = j + 1;
            end
            if j <= length(formants)
                formants(i) = formants(j);
            end
        end
    end
```

```
end
%---------------End of the formant function-------------------------------%
%--------------End of the Linear Predictive Speech Synthesizer------------%
```