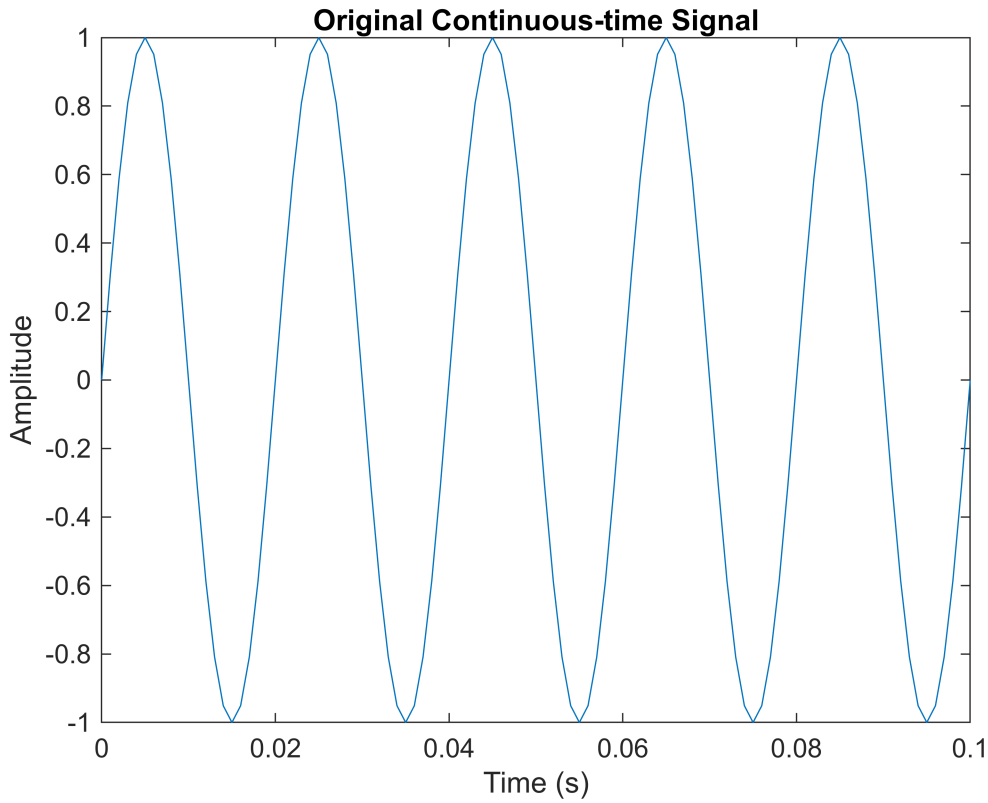
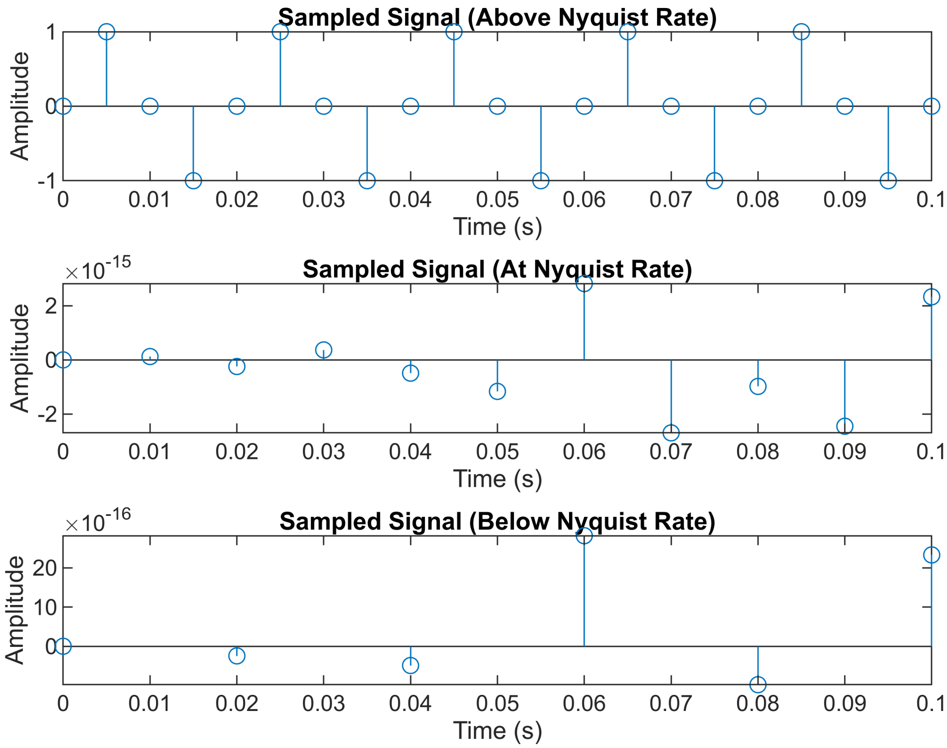
# Part 1

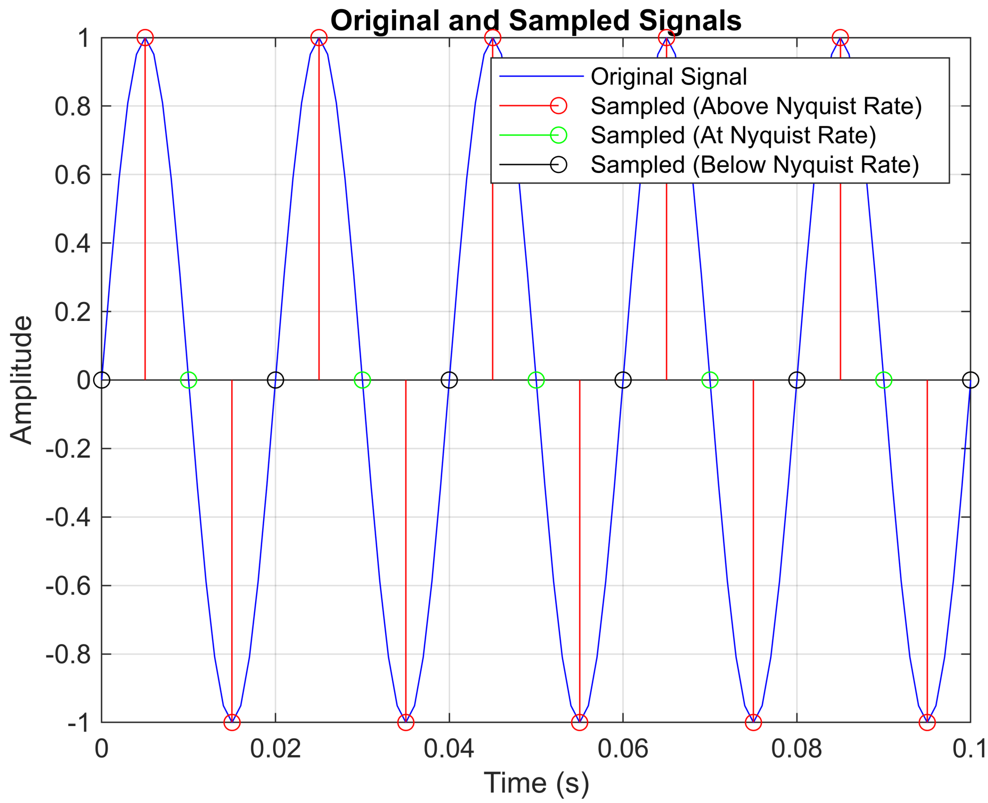
```
% Define parameters
fs_orig = 1000; % Original sampling frequency (Hz)
f_signal = 50;  % Signal frequency (Hz)
t = 0:1/fs_orig:0.1; % Time vector
x = sin(2*pi*f_signal*t); % Continuous-time signal

% Sampling frequencies
fs1 = 200; % Above Nyquist rate
fs2 = 100; % At Nyquist rate
fs3 = 50;  % Below Nyquist rate

t1 = 0:1/fs1:0.1;
x1 = sin(2*pi*f_signal*t1);

t2 = 0:1/fs2:0.1;
x2 = sin(2*pi*f_signal*t2);

t3 = 0:1/fs3:0.1;
x3 = sin(2*pi*f_signal*t3);

figure;
plot(t, x);
title('Original Continuous-time Signal');
xlabel('Time (s)');
ylabel('Amplitude');
```




```
figure;
subplot(3,1,1);
stem(t1, x1);
title('Sampled Signal (Above Nyquist Rate)');
xlabel('Time (s)');
ylabel('Amplitude');

subplot(3,1,2);
stem(t2, x2);
title('Sampled Signal (At Nyquist Rate)');
xlabel('Time (s)');
ylabel('Amplitude');

subplot(3,1,3);
stem(t3, x3);
title('Sampled Signal (Below Nyquist Rate)');
xlabel('Time (s)');
ylabel('Amplitude');
```

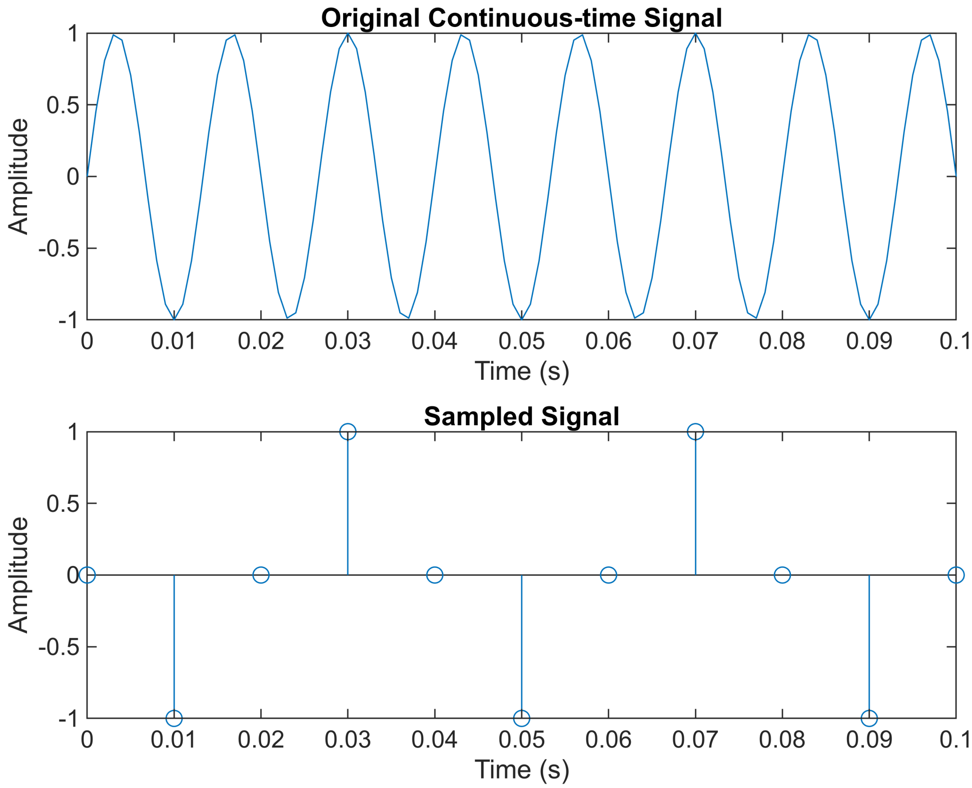



```
figure;
plot(t, x, 'b-');
hold on
stem(t1, x1, 'r-');
stem(t2, x2, 'g-');
stem(t3, x3, 'k-');
title('Original and Sampled Signals');
xlabel('Time (s)');
ylabel('Amplitude');
legend('Original Signal', 'Sampled (Above Nyquist Rate)', 'Sampled (At Nyquist
Rate)', 'Sampled (Below Nyquist Rate)');
grid on
```

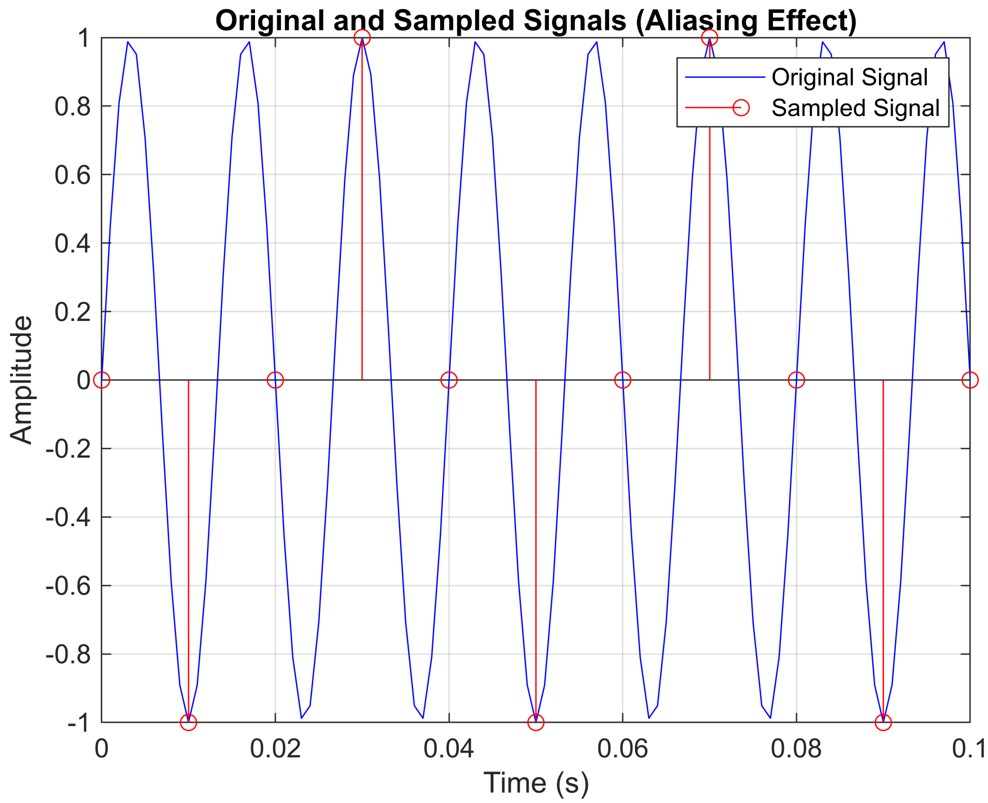



## Part 2

```
% Define parameters
fs = 100; % Sampling frequency (Hz)
f_signal = 75; % Signal frequency (Hz) > fs/2
t = 0:1/1000:0.1; % Time vector
x = sin(2*pi*f_signal*t); % Continuous-time signal

% Sampled signal
n = 0:1/fs:0.1;
x_sampled = sin(2*pi*f_signal*n);

figure;
subplot(2,1,1);
plot(t, x);
title('Original Continuous-time Signal');
xlabel('Time (s)');
ylabel('Amplitude');

subplot(2,1,2);
stem(n, x_sampled);
title('Sampled Signal');
xlabel('Time (s)');
ylabel('Amplitude');
```



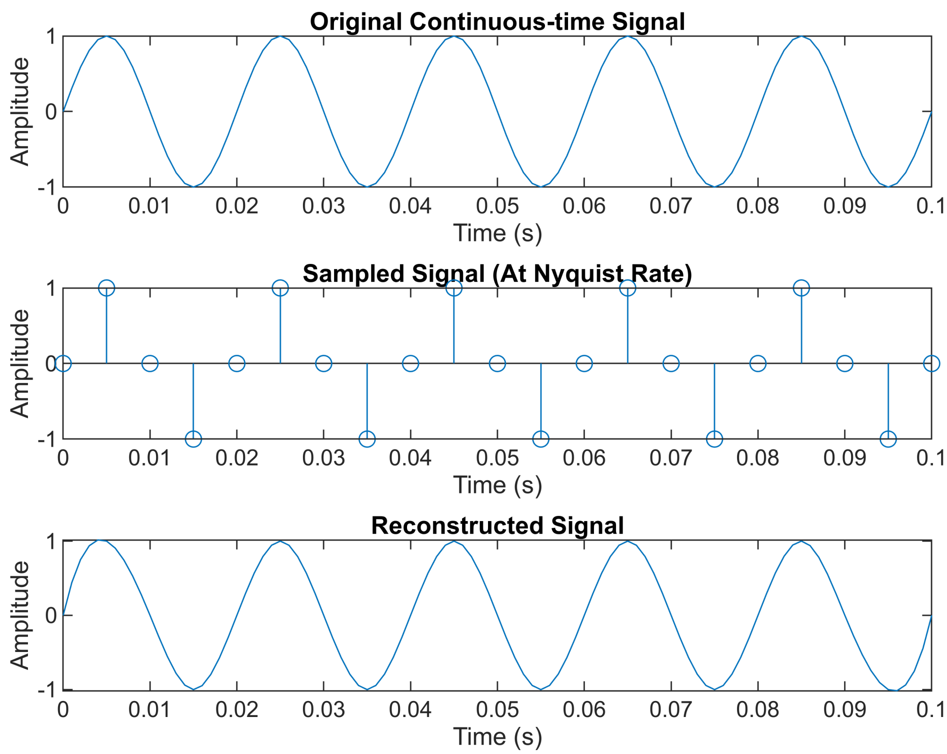

```
figure;
plot(t, x, 'b-');
hold on;
stem(n, x_sampled, 'r-');
title('Original and Sampled Signals (Aliasing Effect)');
xlabel('Time (s)');
ylabel('Amplitude');
legend('Original Signal', 'Sampled Signal');
grid on
```



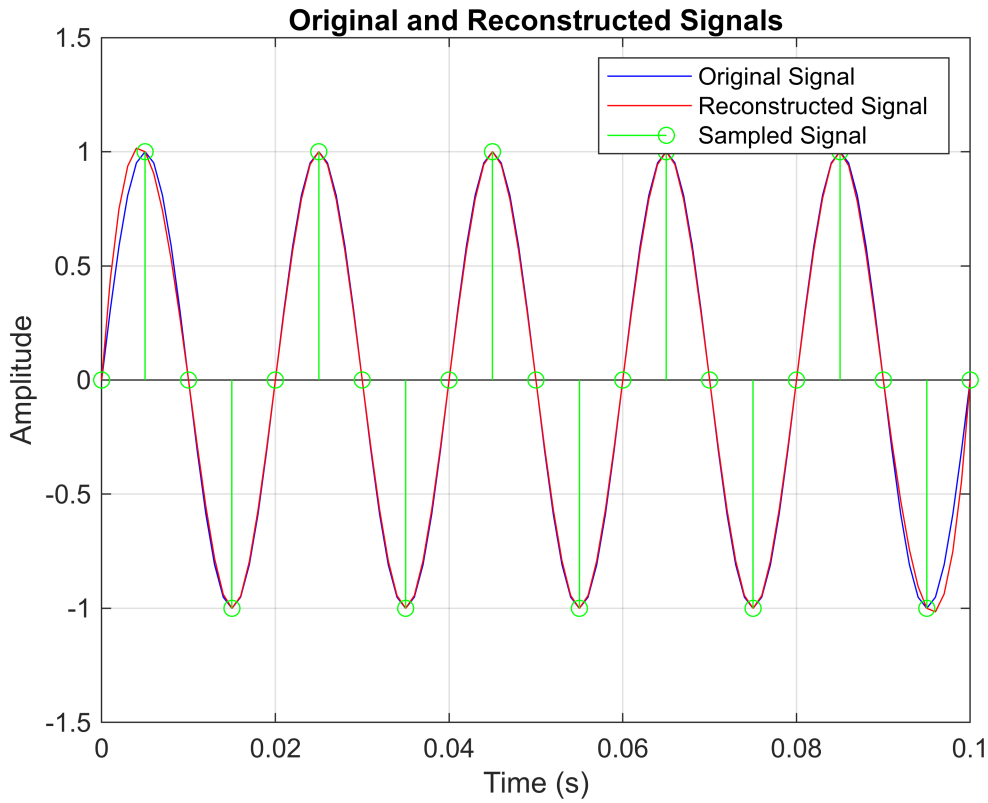

## Part 3

```
% Define parameters
fs = 200; % Sampling frequency (Hz)
f_signal = 50; % Signal frequency (Hz)
t = 0:1/1000:0.1; % Time vector
x = sin(2*pi*f_signal*t); % Continuous-time signal

% Sampled signal
n = 0:1/fs:0.1;
x_sampled = sin(2*pi*f_signal*n);

% Reconstruction
t_recon = 0:1/1000:0.1; % Fine time vector
x_recon = interp1(n, x_sampled, t_recon, 'spline');

figure;
subplot(3,1,1);
plot(t, x);
title('Original Continuous-time Signal');
xlabel('Time (s)');
ylabel('Amplitude');

subplot(3,1,2);
stem(n, x_sampled);
title('Sampled Signal (At Nyquist Rate)');
```

```
xlabel('Time (s)');
ylabel('Amplitude');

subplot(3,1,3);
plot(t_recon, x_recon);
title('Reconstructed Signal');
xlabel('Time (s)');
ylabel('Amplitude');
```




```
figure;
plot(t, x, 'b-', t_recon, x_recon, 'r-');
hold on
stem(n, x_sampled, 'g-');
title('Original and Reconstructed Signals');
xlabel('Time (s)');
ylabel('Amplitude');
legend('Original Signal', 'Reconstructed Signal', 'Sampled Signal');
grid on
```




## Questions

```
% Explain the Nyquist sampling theorem and its significance.
%    The Nyquist sampling theorem states that a continuous-time signal can be
perfectly reconstructed from
%    its samples if the sampling rate is at least twice the highest frequency
present in the signal.
%    This minimum sampling rate is known as the Nyquist rate. The significance of
the Nyquist theorem is that
%    it provides the necessary and sufficient condition for avoiding information
loss during the sampling process.
%    If the signal is sampled at or above the Nyquist rate, the original signal can
be perfectly recovered from the samples.

% Describe the effect of sampling below the Nyquist rate.
%    When a signal is sampled at a rate below the Nyquist rate, a phenomenon called
aliasing occurs.
%    Aliasing happens when high-frequency components in the original signal are
"folded" or "aliased" into
%    lower-frequency components in the sampled signal. This results in the sampled
signal appearing to have
%    a different frequency content than the original signal. Aliasing causes
distortion and loss of information,
%    and the original signal cannot be perfectly reconstructed from the samples.

% What are the limitations of sinc interpolation in signal reconstruction?
```

```
%    Sinc interpolation, also known as ideal low-pass interpolation, is a method
used to reconstruct a continuous-time
%    signal from its samples. While sinc interpolation can provide an exact
reconstruction of the original signal when
%    the sampling rate is at or above the Nyquist rate, it has some limitations:
%    - Sinc interpolation requires an infinite number of samples to perfectly
reconstruct the signal, which is not
%      practical in real-world applications.
%    - The sinc function has a slow decay rate, which means that the reconstructed
signal can exhibit ringing artifacts
%      or Gibbs phenomena near sharp transitions or discontinuities in the original
signal.
%    - Sinc interpolation is sensitive to noise and quantization errors in the
sampled data, which can degrade
%      the quality of the reconstructed signal.

% How can aliasing be avoided in practical applications?
%    To avoid aliasing in practical applications, the following strategies can be
employed:
%    - Ensure that the sampling rate is at least twice the highest frequency
present in the signal (the Nyquist rate)
%      or higher. This can be achieved by using an appropriate sampling device or
by applying an anti-aliasing filter
%      before sampling.
%    - Use an anti-aliasing filter, which is a low-pass filter that removes
frequency components above the Nyquist
%      frequency before sampling. This prevents high-frequency components from
being aliased into the sampled signal.
%    - Oversample the signal, meaning that the sampling rate is significantly
higher than the Nyquist rate.
%      This provides a "buffer" against aliasing and allows for more effective
filtering and reconstruction of
%      the original signal.
%    - Apply digital signal processing techniques, such as decimation and
interpolation, to resample the signal at
%      the desired rate while avoiding aliasing.
```