Problem 1 (Overlap Add and Save Methods). Consider an $L$-point input sequence $x[n] = \text{rand}(1, L)$ and a $P$-point impulse response $h[n] = \begin{cases} 10 & 0 \leq n < P - 1 \\ 0 & \text{otherwise} \end{cases}$

i) Use the following code in MATLAB to compute $y[n] = x[n] \ast h[n]$ for $L = 100000$ and $P = 20$. How long does this computation take?

```matlab
% L is length of input signal and P is length of impulse response.
L=100000, P=20;
x=rand(1,L); % Generate random sequence of numbers between [0,1].
h=10./((0:P-1)+10);
% Direct convolution
tic;
y1=conv(x,h); % Computing convolution of x[n] and h[n].
toc % computing the elapsed time.
```

ii) Segment $x[n]$ to sections of length $B = 50$ as follows:

$$x[n] = \sum_{r=0}^{\infty} x_r[n - rB],$$

where

$$x_r[n] = \begin{cases} x[n + rB] & 0 \leq n \leq B - 1 \\ 0 & \text{otherwise} \end{cases}$$

and show theoretically that $y[n] = \sum_{r=0}^{\infty} y_r[n + rB]$ where $y_r[n] = x_r[n] \ast h[n]$.

iii) Use the following code which is the algorithm mentioned in part (ii) to compute $y[n]$. How long does it take? Compare the resulted elapsed time with the one in part (i).

```matlab
% Overlap-add method :
y2=zeros(SIZE); % Replace the variable SIZE with an appropriate number.
temp=zeros(SIZE);
B=50;
tic;
for i=1:(L/B)
    % each time we consider a B-points window of x, convolve
    % it with h and save the output in B+(P-1) points of temp
    temp( (i-1)*B + 1 : i*B + (P-1) )=conv(x( (i-1)*B + 1 : i*B ),h);
    % add with previous results considering overlaps
    y2=y2+temp;
    % make temp zero
    temp=zeros(SIZE));
end
toc
```
iv) Verify that if a $B$-point sequence is circularly convolved with a $P < B$-point sequence ($P < L$), then the first $P - 1$ points of the result are the only points different from what would be obtained by employing linear convolution.

v) Again divide $x[n]$ into sections of length $B$ so that each input section $x_r[n]$ overlaps the preceding section by $P - 1$ points. Call the circular convolution of each segment with $h[n]$, name it $y_{rp}[n]$. Write $y[n] = x[n] \ast h[n]$ in terms of $y_{rp}[n]$.

Hint: $x_r[n] = x[n + r(B - P + 1) - P + 1], 0 \leq n \leq B - 1$.

vi) Use part (v) to compute $y[n]$ for $L = 100000, P = 20, B = 50$. How long does this take?

**Problem 2.**

i) Let $H(e^{j2\pi f})$ be the ideal low pass filter with cut-off frequency $f_c$, i.e.

$$H(e^{j2\pi f}) = \begin{cases} 
1 & |f| \leq f_c \\
0 & otherwise
\end{cases}$$

Prove that $h[n] = \frac{\sin(2\pi f_c n)}{\pi n}$ for $n \in \mathbb{Z}$.

ii) The following function is a low pass filter with the aforementioned impulse response with cut-off frequency $f_c$ and length $N$:

```matlab
function H = LowPass(fc, N)
    if (N/2 == floor(N/2)) % Check N is even
        h = [sin(2*fc*pi*[1:N/2])./(1:N/2*pi),
             2*fc, sin(2*fc*pi*[1:N/2-1])./(1:N/2-1*pi)];
    else % N is odd
        h = [sin(2*fc*pi*[1:(N-1)/2])./(1:(N-1)/2*pi),
             2*fc, sin(2*fc*pi*[1:(N-1)/2-1])./(1:(N-1)/2-1*pi)];
    end
    H = fft(h, N);
end
```

The following code plots the magnitude and phase of the frequency response of the above filter for $f_c = 0.2$ and $N = 100$. Explain why it is not an ideal low pass? Does it tend to an ideal filter by increasing $N$?

Note that `fftshift(H)` rearranges the outputs of `fft` to the standard form.

```matlab
H = LowPass(0.2, 100);
subplot(2,1,1)
plot((-N/2:N/2-1)/N, abs(fftshift(H)));
subplot(2,1,2)
stem((-N/2:N/2-1)/N, angle(fftshift(H)));
```

iii) Consider the signal $x_0[n] = \sin(0.05n) + \sin(.1n) + \sin(.2n) + \sin(n)$ for $1 \leq n \leq 100$.

Write a code to filter the frequencies larger than 0.1 Hz. Compare your result with $x_{lp}[n] = \sin(0.05n) + \sin(.1n) + \sin(.2n)$. They should be the same.
iv) (Decimation and Interpolation) Consider a system which performs down sampling by factor $M$, i.e. $x_d[n] = x_{lp}[Mn]$, and then performs up sampling by factor $L$, i.e.

$$x_u[n] = \begin{cases} 
x_d[n/L] & \text{if } n = kL \text{ for } k \in \mathbb{Z} \\
0 & \text{otherwise}
\end{cases}$$

First assume that $M = L = 2$. We want to retrieve $x_{lp}[n]$ from $x_u[n]$. Find an appropriate interpolation function ($h_{int}[n]$) such that $x_{lp}[n] = h_{int}[n] \ast x_u[n]$.

Hint: Compare the spectrum of $x_u[n]$ and $x_{lp}[n]$.

v) (Aliasing) Find necessary conditions on $M$ and $L$ in order to successfully retrieve $x_{lp}[n]$ from $x_u[n]$. 