Part 1

MATLAB code of SUMCS function:

```
function [xs] = SUMCS(t,A,omega)
xs = zeros(size(t));
for i = 1 : length(A)
xs = xs + A(i)*exp(1j*omega(i)*t);
end
end
```

MATLAB code of *xs(t):*

```
t = 0:0.001:1 ;
n = mod(22103444,41);
A = (3*rand(1,n)) + (3*rand(1,n)*1j);
omega = pi*rand(1,n);
xs = SUMCS(t,A,omega);
real_xs = real(xs);
imag_xs = imag(xs);
mgn_xs = abs(xs);
phase_xs = angle(xs);

%plot
figure
tiledlayout(2,2)

nexttile
plot(t,real_xs)
title("Real part of xs")
ylabel("Re[xs]")
xlabel("t")

nexttile
plot(t,imag_xs)
title("Imaginer part of xs")
ylabel("Im[xs]")
xlabel("t")

nexttile
plot(t,mgn_xs)
title("Magnitude of xs")
ylabel("Magnitude")
xlabel("t")

nexttile
plot(t,phase_xs)
title("Phase of xs")
ylabel("Phase")
xlabel("t")
```

Part 3

MATLAB code for FSWave:

```
function [xt] = FSWave(t,K,T,W)
xk = zeros(1,2*K+1);
omega = zeros(1,2*K+1);
for k = -K:K
if k == 0
xk(K+1) = (1/T)*(W-W^3/6); %X0
else
xk(k+K+1) = ((2-W^2)/(pi*k) + (T^2)/(pi^3*k^3))*sin((pi*W*k)/T) ...
```

```
- ((W*T)/(pi^2*k^2))*cos((pi*W*k)/T);
end
omega(k+K+1) = 2*pi*k/T;
end
xt = SUMCS(t,xk,omega);
```

MATLAB code for $\tilde{x}(t)$:

```
D11 = mod(22103444,11);
D5 = mod(22103444,5);
K = 20 + D11;
T = 2;
W = 1;
t= -5:0.001:5;
xt = FSWave(t,K,T,W);
k = -20:20;
real_xt = real(xt);
imag_xt = imag(xt);

figure
tiledlayout(1,2)

nexttile
plot(t,real_xt)
title("Real part of xt")
ylabel("Re[xt]")
xlabel("t")

nexttile
plot(t,imag_xt)
title("Imaginary part of xt")
ylabel("Im[xt]")
xlabel("t")

max_real = max(real_xt);
min_real = min(real_xt);
max_imag = max(imag_xt);
min_imag = min(imag_xt);
```

Part 4

Part a

```
function [xt] = FSWave(t,K,T,W)
Xk = zeros(1,2*K+1);
omega = zeros(1,2*K+1);
for k = -K:K
k = -k;
if k == 0
Xk(K+1) =(1/T)*(W-W^3/6); %X0
else
Xk(k+K+1) = (((2-W^2)/(2*pi*k) + (T^2)/(pi^3*k^3))*sin((pi*W*k)/T) ...
- ((W*T)/(pi^2*k^2))*cos((pi*W*k)/T));
end
omega(k+K+1) = 2*pi*k/T;
end
xt = SUMCS(t,Xk,omega);
end
```

Part b

```
function [xt] = FSWave(t,K,T,W)
Xk = zeros(1,2*K+1);
```

```
omega = zeros(1,2*K+1);
t0 = 0.6;
for k = -K:K
if k == 0
Xk(K+1) =(1/T)*(W-W^3/6); %X0
else
Xk(k+K+1) = (((2-W^2)/(2*pi*k) + (T^2)/(pi^3*k^3))*sin((pi*W*k)/T) ...
- ((W*T)/(pi^2*k^2))*cos((pi*W*k)/T));
Xk(k+K+1) = (Xk(k+K+1))*(exp(-1i*2*pi*k*t0/T));
end
omega(k+K+1) = 2*pi*k/T;
end
xt = SUMCS(t,Xk,omega);
end
```

Part c

```
function [xt] = FSWave(t,K,T,W)
Xk = zeros(1,2*K+1);
omega = zeros(1,2*K+1);
for k = -K:K
k = -k;
if k == 0
Xk(K+1) =(1/T)*(W-W^3/6); %X0
else
Xk(k+K+1) = (((2-W^2)/(2*pi*k) + (T^2)/(pi^3*k^3))*sin((pi*W*k)/T) ...
- ((W*T)/(pi^2*k^2))*cos((pi*W*k)/T));
Xk(k+K+1) = (Xk(k+K+1))*(1i*k*2*pi/T);
end
omega(k+K+1) = 2*pi*k/T;
end
xt = SUMCS(t,Xk,omega);
end
```

Part d

```
function [xt] = FSWave(t,K,T,W)
Xk = zeros(1,2*K+1);
omega = zeros(1,2*K+1);
for k = -K:K
k = -k;
if k == 0
Xk(K+1) =(1/T)*(W-W^3/6); %X0
else
Xk(k+K+1) = (((2-W^2)/(2*pi*k) + (T^2)/(pi^3*k^3))*sin((pi*W*k)/T) ...
- ((W*T)/(pi^2*k^2))*cos((pi*W*k)/T));
end
omega(k+K+1) = 2*pi*k/T;
end

negative=Xk(1:K);
negative=[flip(negative),zeros(1,K+1)];
positive=Xk(K+2:2*K+1);
positive=[zeros(1,K+1),flip(positive)];
Xk=negative+positive;
xt = SUMCS(t,Xk,omega);
end
```