Name: Manaal Waseem **Reg. No:** FA18-BCE-074

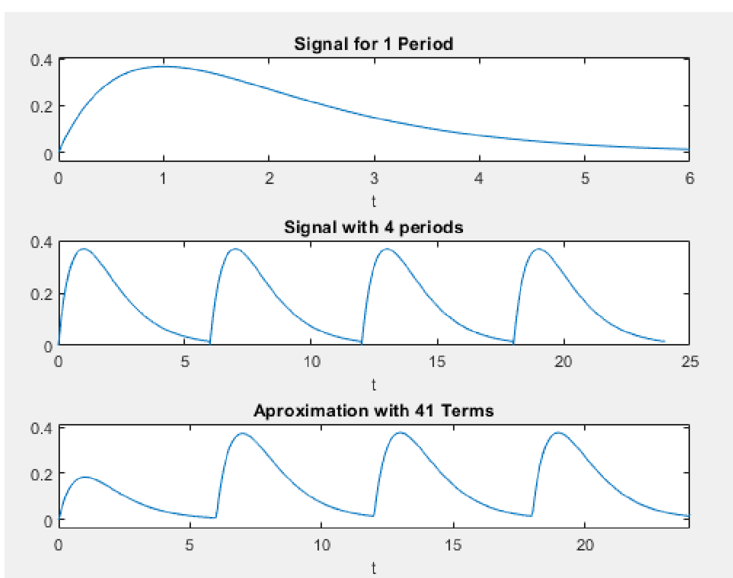
Lab 9

Task 1:

Code:

```
t0 = 0;
T = 6;
w = 2.*pi./T;
syms t
x = t.*exp(-t);
subplot(3,1,1)
ezplot(x, [t0 t0+T])  % plots f over the specified
% ranges along the abscissa and the ordinate.
title('Signal for 1 Period')
t1 = t0:0.01:T;
xx = t1.*exp(-t1);
X = repmat(xx,1,4); % repmat(A,m,n) creates a large
% matrix B consisting of an m -by- n tiling of copies of A
tt = linspace(0,4.*T,length(X)); %  generates linearly
% spaced vectors
subplot(3,1,2)
plot(tt,X)
xlabel('t')
title('Signal with 4 periods')
for k = -40:40
a(k+41) = (1/T).*int(x.*exp(-1i.*k.*w.*t), t, t0, t0+t);
end
for k = -40:40
ex(k+41) = exp(1i.*k.*w.*t);
end
xx1 = sum(a.*ex);
subplot(3,1,3)
ezplot(xx1, [t0 t0+4*T])
title('Aproximation with 41 Terms')

```

Graph:

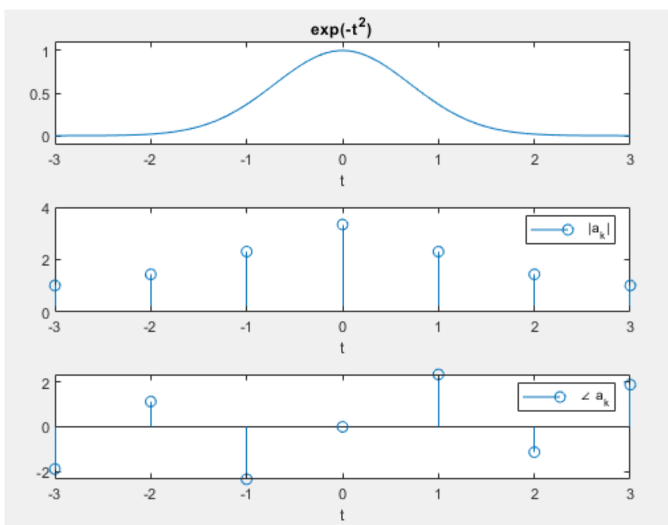


Task 2:

Code:

```
t0 = -3;
T = 6;
w = 2.*pi./T;
syms t
x = exp(-t.^2);
subplot(3,1,1)
ezplot(x, [t0 t0+T]) % plots f over the specified
% ranges along the abscissa and the ordinate.
xlabel('t')
syms t k n
x = exp(-t);
a = (1/T)*int(x.*exp(-1i.*k.*w.*t), t, t0, t0+T);
k1 = -3:3;
A = subs(a, k, k1); %subs(s,old,new) returns a copy
%of s, replacing all occurrences of old with new,
%and then evaluates s.
subplot(3,1,2)
stem(k1,abs(A));
legend('|a_k|')
xlabel('t')
subplot(3,1,3)
stem(k1,angle(A)); % returns the phase angles
legend('\angle a_k')
xlabel('t')
```

Graph:



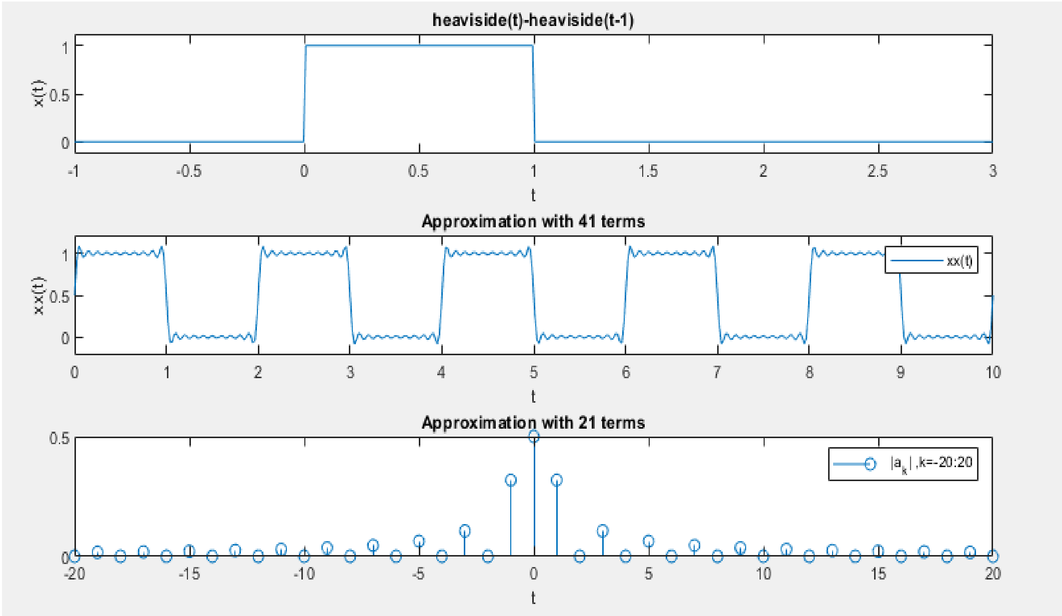
Task 3:

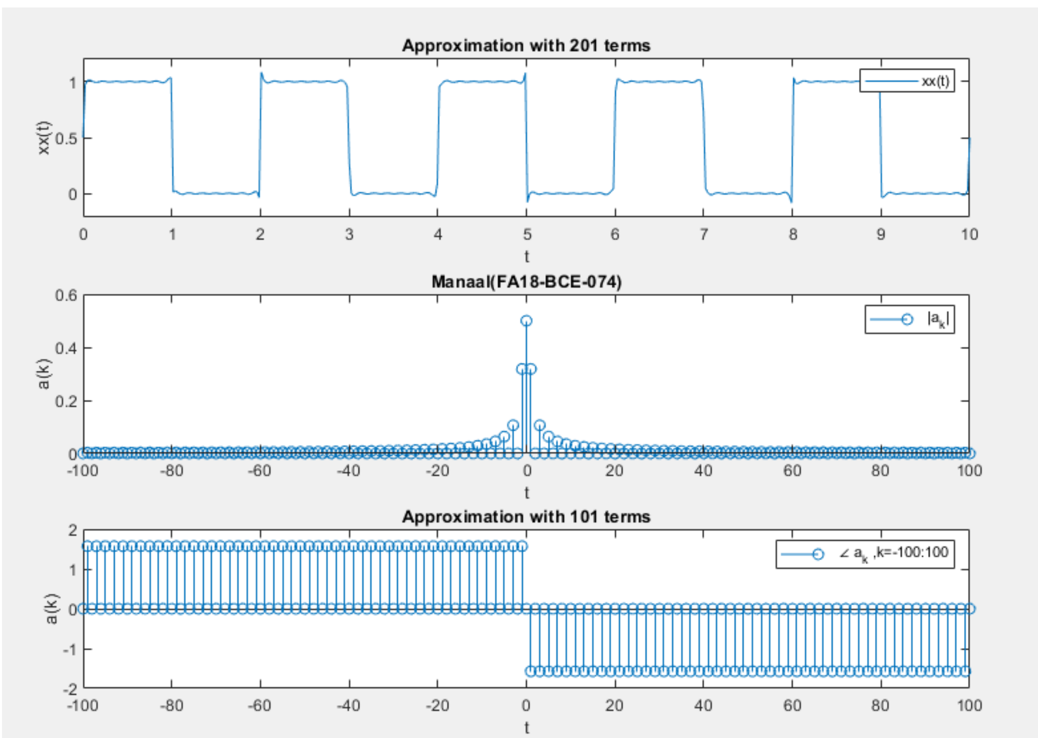
Code:

```
t0 = 0;
T = 2;
w = 2*pi/T;
syms t n k
x=heaviside(t)-heaviside(t-1);
subplot(4,1,1)
ezplot(x, [-1 3])
title('heaviside(t)-heaviside(t-1)')
xlabel('t')
ylabel('x(t)')

k=-20:20;
t0 =0;
T=2;
w=2*pi/T;
a=(1/T)*int(x*exp(-j*k*w*t),t,t0,t0+T)
xx=sum(a.*exp(j*k*w*t))
subplot(4,1,2)
ezplot(xx,[0 10])
title('Approximation with 41 terms')
legend('xx(t)')
xlabel('t')
ylabel('xx(t)')

a1=eval(a)
subplot(4,1,3)
stem(-20:20,abs(a1));
legend ('|a_k| ,k=-20:20')
title('Approximation with 21 terms')
xlabel('t')
```

```
k=-100:100;
t0 =0;
T=2;
a=(1/T)*int(x*exp(-j*k*w*t),t,t0,t0+T);
xx=sum(a.*exp(j*k*w*t));
subplot(3,1,1)
ezplot(xx,[0 10])
title('Approximation with 201 terms')
legend('xx(t)')
xlabel('t')
ylabel('xx(t)')

a1=eval(a)
subplot(3,1,2)
stem(-100:100,abs(a1));
legend ('|a_k|')
title('Manaal(FA18-BCE-074)')
xlabel('t')
ylabel('a(k)')
subplot(3,1,3)
stem(-100:100,angle(a1));
legend ('\angle a_k ,k=-100:100')
title('Approximation with 101 terms')
xlabel('t')
ylabel('a(k)')
```

Graph:


heaviside(t)-heaviside(t-1)
x(t)
t
Approximation with 41 terms
xx(t)
Approximation with 21 terms
|a_k| ,k=-20:20


Approximation with 201 terms
xx(t)
t
Manaal(FA18-BCE-074)
a(k)
|a_k|
Approximation with 101 terms
∠ a_k ,k=-100:100

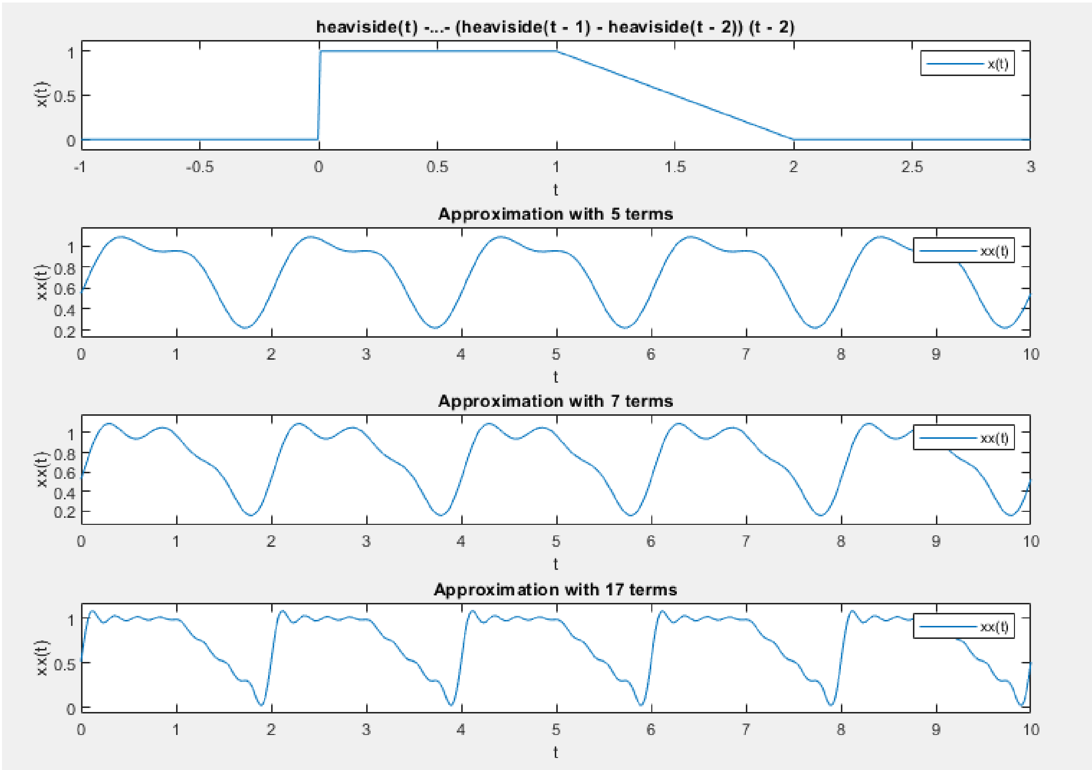
Task 4:

Code:

```
t0 = 0;
T = 2;
w = 2*pi/T;
syms t n k
x=heaviside(t)-heaviside(t-1)+(2-t)*(heaviside(t-1)-heaviside(t-2))
subplot(4,1,1)
ezplot(x, [-1 3])
legend('x(t)')
xlabel('t')
ylabel('x(t)')

k=-2:2;
t0 =0;
T=2;
w=2*pi/T;
a=(1/T)*int(x*exp(-j*k*w*t),t,t0,t0+T)
xx=sum(a.*exp(j*k*w*t))
subplot(4,1,2)
ezplot(xx,[0 10])
title('Approximation with 5 terms')
legend('xx(t)')
xlabel('t')
ylabel('xx(t)')

k=-3:3;
t0 =0;
T=2;
w=2*pi/T;
a=(1/T)*int(x*exp(-j*k*w*t),t,t0,t0+T)
```

```
xx=sum(a.*exp(j*k*w*t))
subplot(4,1,3)
ezplot(xx,[0 10])
title('Approximation with 7 terms')
legend('xx(t)')
xlabel('t')
ylabel('xx(t)')

k=-8:8;
t0 =0;
T=2;
w=2*pi/T;
a=(1/T)*int(x*exp(-j*k*w*t),t,t0,t0+T)
xx=sum(a.*exp(j*k*w*t))
subplot(4,1,4)
ezplot(xx,[0 10])
title('Approximation with 17 terms')
legend('xx(t)')
xlabel('t')
ylabel('xx(t)')

```

Graph:



Critical Analysis:

In this lab I learnt:

- The complex coefficients a_k are called complex exponential Fourier series coefficients, while a_0 is a real number and is called a constant or dc component.
- Each coefficient a_k corresponds to the projection of the signal $x(t)$ at the frequency $k\Omega_0$, which is known as k^{th} harmonic.
- Fourier series coefficients represent the signal in the frequency domain, they are also referred as the spectral coefficients of the signal.
- I learnt to analyze/decompose a continuous time signal into frequency components given by sinusoidal signals.
- This process in crucial in the signal processing field since it reveals the frequency content of signal and simplifies the calculation of systems' output.

There are three different and equal ways that can be used in order to express a signal into sum of simple oscillating functions, i.e., into a sum of sines, cosines, or complex exponentials.

THE END