***2.1 – Function for Restoring x(t)using the Fourier coefficients:***

function y = partial(A,T,t)

y=zeros(size(t));

%declaring a row vector of size (1,no. of values in t)

N=(length(A)-1)/2;

%finding the no. of fourier coefficients from A

for k=-N:N

y=y+A(k+N+1)\*exp(1i\*k\*(2\*pi/T)\*t);

%calcuating y(t) using ak\*e^(jWot)

end

plot(t,real(y));

%plotting the graph of real(y(t)) vs t

title ("Construction of signal from Fourier Coefficients");

xlabel("time")

ylabel("y(t)")

end

***2.2 – Function for generating fourier coefficients of rectangular pulse and an ideal rectangular pulse:***

function [A,y,sq] = squarefs(T,T1,t,N)

A=zeros(1,2\*N+1);

%initializing a row vector of size 2\*N+1 with all zeros

for k = -N:N

%if k==0 manually calculated a0=integral(y(t))over the time period

if(k==0)

A(k+N+1)=2\*T1/T;

else

%calculating ak using the sin(kWoT1)/(k\*pi)

A(k+N+1)=sin(k\*T1\*2\*pi/T)/(k\*pi);

end

end

y = zeros(size(t));

for k=-N:N

y=y+A(k+N+1)\*exp(1i\*k\*(2\*pi/T)\*t);

%calcuating y(t) using ak\*e^(jWot)

end

y = real(y);

sq=rectangularPulse(-T1,T1,t);

%calculating rectangular pulse using an inbuilt function

end

***2.2 a – Script for calculating and plotting Fourier Transform of Square wave with different Time Periods:***

t=-0.5:0.01:0.5;

%time from -0.5 to 0.5sec with a difference 0.01

[A,y,sq] = squarefs(0.2,0.1,t,100) ;

subplot(5,1,1);

plot(-100:100,A,'b');

title("Fourier Series coefficients vs 'k'");

xlabel('k(-100:100)')

ylabel("a\_{k} for T = 0.2sec")

%plotting the graph of ak vs k for T=0.2sec

for T = 0.40:0.20:1

[A,y,sq] = squarefs(T,0.1,t,100) ;

subplot(5,1,T\*5);

plot(-100:100,A,'b');

xlabel('k(-100:100)')

ylabel(["a\_{k} for T = " T "sec"])

%plotting the graph of ak vs k for T=0.4sec to T=1sec

end

***2.2b – Script for comparing and plotting the ideal rectangular pulse sq(t) and Fourier series of the rectangular pulse y(t):***

t=-0.5:0.01:0.5;

%time from -0.5 to 0.5sec with a difference 0.01

subplot(2,1,1);

plot(t,y,'r');

title('Square Function reconstructed from Fourier Series N=100')

xlabel('time')

ylabel('y(t)')

% plotting graph of y(t) vs t

subplot(2,1,2);

plot(t,sq,'b');

title('Original Square Function')

xlabel('time')

ylabel('sq(t)')

% plotting graph of sq(t) vs t

***2.3 – Script for computing errors:***

err0 = zeros(1,100);

%making a row vector with 100 zeros

err1 = zeros(1,100);

%making a row vector with 100 zeros

t=-0.5:0.001:0.5;

%time from -0.5 to 0.5 with difference of 0.001

for k = 1:100

[A,y,sq] = squarefs(1,0.1,t,k) ;

err0(k) = max(abs(y-sq));

%for calculating maximum of the absolute values of the error

err1(k)=sum((y-sq).\*(y-sq))/length(y);

%for calculating mean squared values of the error

end

subplot(2,1,1);

plot(1:100,err0,'r');

%plotting the absolute error vs no. of terms in fourier series

%representation of signal

title('Maximum Absolute Error')

xlabel('No. of terms in Fourier series')

ylabel('Absolute error')

subplot(2,1,2);

plot(1:100,err1,'b');

%plotting the mean squared error vs no. of terms in fourier series

%representation of signal

title('Mean Squared Error')

xlabel('No. of terms in Fourier series')

ylabel('Absolute error')