



clear;

clc;

close all;

% Newton-Raphson Method

% f(x) = cos(x) - x

xn=1;

error = 0.0000001; %percent

counter =0;

while (true)

xp=xn;

counter = counter +1;

xn = (xn-(cos(xn)-xn)/(-1\*sin(xn)-1)) ;

h = xn-xp;

h\_vector(1,counter) = h ;

error\_new = abs( (xn-xp)/xn ) \* 100 ;

error\_vector(1,counter) = error\_new;

if error\_new < error

break

end

end

fprintf('This is the root for cos(x) - x. : %d', xn)

x = log(1./h\_vector) ;

y = log(error\_vector);

figure()

plot(x,y)

xlabel('log(1/h)')

ylabel('log(error)')

title('cos(x) -x')

% f(x) = e^(-x) - x

xn = 1 ;

error = 0.00000001;

counter =0;

h\_vector= [];

error\_vector = [];

while (true)

xp = xn;

counter = counter +1;

xn = xn - ((exp(-xn) - xn) / (-exp(-xn) -1));

h = xn-xp;

h\_vector(1,counter) = h ;

error\_new = abs( (xn-xp)/xn ) ;

error\_vector(1,counter) = error\_new;

if error\_new < error

break

end

end

fprintf('\nThis is the root for e^-x - x : %d ', xn)

x = log(1./h\_vector) ;

y = log(error\_vector);

figure()

plot(x,y)

xlabel('log(1/h)')

ylabel('log(error)')

title('e^-x -x ')