**MATH 333: Numerical Analysis**

**Lab 6: Newton Raphson method for system of non linear equation & Curve fitting**

# Task 1:

## Lab\_6.m

a = input('Enter lower bound of interval:');

b = input('Enter upper bound of interval:');

tol = input('Enter tolerance:');

%a = 4;

%b = 6;

%tol = [0.01; 0.01];

X0 = [a; b];

iter = 0;

for i = 2:10

iter=iter+1;

X1 = X0 - J(a,b)\*F1(a,b)

x1 = X1(1,1);

x2 = X1(2,1);

F = F1(x1,x2)

fprintf('Iteration %d: x1 = %.2f x2 = %.2f\n\n',iter, x1, x2)

if abs(F1(x1,x2)) <= tol

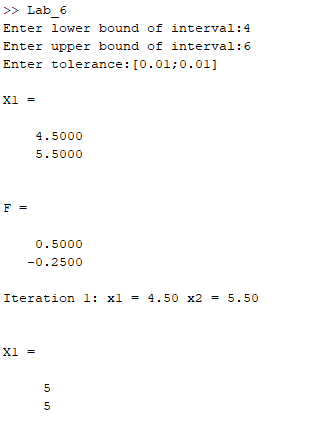
fprintf('DONE! Iteration %d: x1 = %.4f x2 = %.4f\n\n',iter, x1, x2)

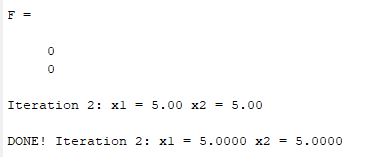
break;

end

X0 = X1;

end





## J.m

function[a] = J(x1, x2)

syms x y

f1 = x.^2 + y.^2 -50;

f2 = x.\*y -25;

j = [diff(f1,x) diff(f1,y); diff(f2,x) diff(f2,y)];

x = x1;

y = x2;

a = eval(inv(j));

## F1.m

function[a] = F1(x1,x2)

syms x

syms y

f1 = x.^2 + y.^2 -50;

f2 = x.\*y -25;

x = x1;

y = x2;

b = eval(f1);

c = eval(f2);

a = [b;c];

# Task 2:

x = [0:8];

y=[40.12 66.78 80.17 86.71 80.77 66.78 44.41 10.51 -32.60];

pc = polyfit(x,y,1); % 1 means linear fit

plot(x,y,'ro');

hold on;

plot(x,polyval(pc,x),'b-');

txt = sprintf('Best fit line y=%.2fx + %.2f',pc(1),pc(2));

legend('Data points',txt);

fprintf('Slope = %.2f\n', pc(1));

fprintf('Intercept = %.2f\n', pc(2));



a = polyval(pc, 4.5);

b = polyval(pc, 8.5);

