Assignment 1

Name – Abhishek Agrahari

Roll Number – 190123066

Question 1-

By Newton's method these system of non-linear equation can be solved. I have taken tolerance to be 10^(-3) and starting value of [x_1,x_2] as [1,2]. For stopping condition I have used abs($h_1^{(n)}$) + abs($h_2^{(n)}$) < TOL = 10^(-3).

Roots obtained are – [x_1, x_2] = [1.08613815, 1.94371377]

Code

```
clc;
clear;
TOL = 10^(-3);

f1 = @(x1,x2) sin(x1*x2) + x1-x2;
f2 = @(x1,x2) x2*cos(x1*x2) + 1;
df1_dx1 = @(x1,x2) x2*cos(x1*x2) + 1;
df1_dx2 = @(x1,x2) x1*cos(x1*x2)-1;
df2_dx1 = @(x1,x2) -x2*x2*sin(x1*x2);
df2_dx2 = @(x1,x2) cos(x1*x2) - x1*x2*sin(x1*x2);
J = @(x1,x2) [df1_dx1(x1,x2) df1_dx2(x1,x2); df2_dx1(x1,x2) df2_dx2(x1,x2)];
h = @(x1,x2) -inv(J(x1,x2))*[f1(x1,x2); f2(x1,x2)];

i = 1;
h1 = zeros(100);
h2 = zeros(100);
X1 = zeros(100);
X2 = zeros(100);
X1(i) = 1;
X2(i) = 2;
H = h(X1(i),X2(i));
h1(i) = H(1,1);
h2(i) = H(2,1);
fprintf("Iteration        x1            x2           f1(x1,x2)        f2(x1,x2)\n");
while abs(h1(i)) + abs(h2(i)) >= TOL
    fprintf(" %5d %14.8f %14.8f %14.8f
%14.8f\n",i,X1(i),X2(i),f1(X1(i),X2(i)),f2(X1(i),X2(i)));
    i = i + 1;
    X1(i) = X1(i-1) + h1(i-1);
    X2(i) = X2(i-1) + h2(i-1);
    H = h(X1(i),X2(i));
    h1(i) = H(1,1);
    h2(i) = H(2,1);
end
    fprintf(" %5d %14.8f %14.8f %14.8f
%14.8f\n",i,X1(i),X2(i),f1(X1(i),X2(i)),f2(X1(i),X2(i)));
```

Ouput

Iteration	x1	x2	f1(x1,x2)	f2(x1,x2)
1	1.00000000	2.00000000	-0.09070257	0.16770633
2	1.07966476	1.94538540	-0.00269513	0.01726846
3	1.08613815	1.94371377	-0.00004455	0.00009109

Question 2

In this question, I have taken starting point as 2 and tolerance to be 10^(-5). Stopping condition I have used is abs(f(x_n)) < TOL = 10^(-5). By output we can observe that standard newton method takes 11 iterations to complete while modified newton method takes only 4 iterations to complete. This is because order of convergence in standard newton method is close to 1 while in modified newton method it is close to 2.

Final order of convergence in standard newton's method = 1.0008309

Final order of convergence in modified newton's method = 1.9785775

Formula used for finding order of convergence –

$$p = \frac{log_{10}(\frac{|e_{n+2}|}{|e_{n+1}\}})}{log_{10}(\frac{|e_{n+1}|}{|e_n\}})}$$

where e_n = 1.1-x_n.

Code

```
clc;
clear all;
f = @(x) ((x-1.1)^2)*(x+1);
f_prime = @(x) ((x-1.1)^2) + 2*(x+1)*(x-1.1);
err = @(x) abs(1.1-x);
TOL = 10^(-5);
x = zeros(1,100);
xm = zeros(1,100);
t = 1;
tm = 1;
x(t) = 2;
xm(tm) = 2;

while abs(f(x(t))) >= TOL
    t = t + 1;
    x(t) = x(t-1) - f(x(t-1))/f_prime(x(t-1));
end

while abs(f(xm(tm))) >= TOL
    tm = tm + 1;
```

```
    xm(tm) = xm(tm-1) - 2*f(xm(tm-1))/f_prime(xm(tm-1));
end

fprintf('  Iteration     Newton Method           p\n')
for k = 1:t
    if k + 2 <= t
        fprintf('%8d %18.7f %18.7f\n',k,x(k), log10(err(x(k+2))/err(x(k+1)))
/log10(err(x(k+1))/err(x(k))))
    else
        fprintf('%8d %18.7f\n',k,x(k))
    end
end

fprintf('  Iteration  Modified Newton Method      p\n')
for k = 1:tm
    if k + 2 <= tm
        fprintf('%8d %18.7f %18.7f\n',k,xm(k), log10(err(xm(k+2))/err(xm(k+1)))
/log10(err(xm(k+1))/err(xm(k))))
    else
        fprintf('%8d %18.7f\n',k,xm(k))
    end
end
```

Ouput

```
  Iteration     Newton Method           p
       1          2.0000000          1.0657109
       2          1.6086957          1.0518454
       3          1.3769436          1.0352990
       4          1.2460946          1.0213418
       5          1.1753481          1.0118888
       6          1.1383154          1.0063007
       7          1.1193278          1.0032472
       8          1.1097078          1.0016489
       9          1.1048650          1.0008309
      10          1.1024353
      11          1.1012184
  Iteration  Modified Newton Method      p
       1          2.0000000          1.7958073
       2          1.2173913          1.9785775
       3          1.1030273
       4          1.1000022
```