1. Consider the following matlab codes for implementation of the Newton method of root finding.

```
s='x.^2-5*x+6'
f=inline(s); x=sym('x')
ss=['diff(' s ') '];
s1=eval(ss);
f1=inline(s1);
x_zero=rand;
while   ~( abs(f(x_zero))< 10^-6)
      x_zero=x_zero-f(x_zero)/f1(x_zero)
end
```

Revise above codes to implement the Newton method based on numerical differentiation of Richardson Extrapolation.

2. Consider the following matlab codes for implementation of Jacobian

```
s1='3*x1-cos(x2*x3)-1/2';
s2='x1^2-81*(x2+0.1)^2+sin(x3)+1.06';
s3='exp(-x1*x2)+20*x3+1/3*(10*pi-3)';
x1=sym('x1');x2=sym('x2');x3=sym('x3');
f=inline([str2sym(s1);str2sym(s2);str2sym(s3)]);
A=jacobian([str2sym(s1);str2sym(s2);str2sym(s3)],[x1 x2 x3]);
j=inline(A);
x=rand(3,1)-0.5;
y=f(x(1),x(2),x(3))
j(x(1),x(2),x(3))
```

Revise above codes to determine Jacobian based on numerical differentiation of Richardson Extrapolation.