```
clear all;
syms x;
f_str = input('Function (i.e x*(x^2+1))','s');
str2func(f_str);
% f = x / (x^2+1);
i_guess = input('Initial Guess');
% i_guess = -0.70;
xf = i_guess;
% Lower bound of range of 'x' to be seen
lxrange = -5.00;
% Upper bound of range of 'x' to be seen
uxrange = 5.00;
g=diff(f);

% The following finds the upper and lower 'y' limits for the plot based on the given
%    'x' range in the input section.
maxi = subs(f,x,lxrange);
mini = subs(f,x,lxrange);
for i=lxrange:(uxrange-lxrange)/10:uxrange
    if subs(f,x,i) > maxi
        maxi = subs(f,x,i);
    end
    if subs(f,x,i) < mini
        mini = subs(f,x,i);
    end
end
tot=maxi-mini;
mini=mini-0.1*tot;
maxi=maxi+0.1*tot;

% This calculates window size to be used in figures
set(0,'Units','pixels')
scnsize = get(0,'ScreenSize');
wid = round(scnsize(3));
hei = round(0.95*scnsize(4));
wind = [1, 1, wid, hei];

% Epsilon should be set a small value
eps = 0.001
% initially xs should be set a value
% that don't inhibit loop
xs = xf + eps + 1;
iter = 0;

while ( abs(xf) > eps | abs(xf - xs) ) & iter < 10
    iter = iter + 1;
    figure('Position',wind)
    clf
    ezplot(f,[lxrange,uxrange])
    hold on
    title('Newton Raphson Method')
    xs=xf-subs(f,x,xf)/subs(g,x,xf);
    m=-subs(f,x,xf)/(xs-xf);
    b=subs(f,x,xf)*(1+xf/(xs-xf));
    lefty=(maxi-b)/m;
    righty=(mini-b)/m;
    iter
    xf
    xs
    plot([xf,xf],[maxi,mini],'g','linewidth',2)
```

```
    plot([lefty,righty],[maxi,mini],'r','linewidth',2)
    plot([lxrange,uxrange],[0,0],'k','linewidth',1)
    plot([lxrange,uxrange],[0,0],'k','linewidth',1)
    xf=xs;
    if ( abs(xf) <  eps )
        fprintf(1, 'f(x) is smaller than epsilon\n', a)
    end
end
hold off
```