Assignment 3
Solutions

1) a)

```
function [x,fx,it,flag] = myNewton(f,df,x0,maxit,tol)
it=0; flag=0; % Flag 0 - not converged.
while (it<maxit)
    it=it+1;
    y=f(x0); dy=df(x0);
    if abs(dy) < eps
        flag=-1; % Flag -1 - derivative become zero.
        break;
    end
    x1 = x0 - y/dy; % Newton step
    if abs(x1-x0) < tol*abs(x1)
        flag = 1; % Flag 1 - converged
        break;
    end
    x0 = x1;
end
x=x1; fx=f(x1);
```

b) Newton's method page of Wikipedia has it covered under "Failure Analysis".

2) a) Derivation should be straightforward - Derivative of f has a root at $x=-1$, but otherwise nonzero and continuous. (Note that, it also has a root a $x=1$, but f has one as well so they will cancel around x.) Since f has a double root at $x=1$, we should see linear convergence.

b) It should converge to $x=1$ at a linear speed. Getting a solution correct up to 10^{-16} should take $-\log_2 10^{-16} \approx 16 \times 3.3 \approx 50$ steps, assuming error halves every step. This might not be observable due to numerical errors or assumption being wrong.

3) a) Assume f has a root of multiplicity m and $f \in C^{m+1}$. Then by Taylor's theorem; $$f(x) = (x-x^*)^m \frac{f^{(m)}(x^*)}{m!} + O((x-x^*)^{m+1})$$

and $$f'(x) = (x-x^*)^{m-1} \frac{f^{(m)}(x^*)}{m!} + O((x-x^*)^m)$$

where x^* is the root of multiplicity m. Direct calculation of w should give the expected answer.

(Here we used; ~~x~~ is a multiple root of f iff $f(a)=f'(a)=0$).

b) Should be straightforward.

c) Advantages

— This method is still second order convergent even though there is a multiple root for f. (Technically eliminates the linear convergence problem of regular Newton's method.

Disadvantage

— Requires second derivatives.

~~Computationally may introduce~~

4) Say x is the root. Define $x_k = x + e_k$, e_k error at iteration k. Find Taylor expansion of $f(x_k)$ around x. Consider $x_{k+1} - x_k$, and using Taylor expansion derive the relationship between e_{k+1} and e_k

5) Using Taylor expansion show ~~each~~ whether each ϕ_j is a contraction mapping for α_1 and α_2.