Modified Secant Method:

uses one point to find the root, and the derivative is found by using another point some small distance, δ, away.

Pseudocode:

Input: function ‘f’, init guess ‘x’, ‘delta’, error tol ‘etol’, max no. of iteration ‘maxit’

output: root x, no. of iteration nit

start

while the actual no. of iteration is still < maxit

xold = x

recalculate the new x where xnew = x – delta*x*f(x)/ (f(x+delta*x)-f(x))

if the actual error tolerance |x – xold| <= etol, end while loop

end

```
function [x, nit] = modsec(f,x,delta,etol,maxit)
% This function finds the root of a function using
% The Midified Secant Method
% INPUT:
% f = function to find root to
% x = initial guess
% delta = perturbation fraction
% etol = error tolerance
% maxit = maximum number of iterations
%
% OUTPUT:
% root = approximation of the root of f
  nit = 0;
  while nit < maxit
   xold = x;
   x = x - delta*x*f(x)/(f(x+delta*x)-f(x));
   nit = nit + 1;
   if abs(x - xold) < etol, break, end
  end
end
```

```
>> [root, nit] = modsec(@(x) x ^ 3 - 3 * x + 1, 0.9, 0.001, 0.00005, 100)
root =  1.5321
nit =  7
```