
```

% -----inputs-----
f = @(x) exp(x)-x-1;
df = @(x) exp(x)-1; % function derivative

p0 = 1;
TOL = 1e-5;
NI = 100;
% -----
fprintf("\nNewton's Method")

% STEP 1
i = 1;
converge = false; % convergence flag

% STEP 2
while i<=NI
    % STEP 3
    % compute p(i)
    p = p0-f(p0)/df(p0);
    err = abs(p-p0);
    % STEP 4
    % check if meets the stopping criteria
    if (err< TOL)
        converge = true;
        break
    else
        % STEP 5
        i = i+1;
        % STEP 6
        p0 = p; % update p0
    end
end

if converge
    fprintf('\n\nApproximate solution P = %.8f\n',p)
    fprintf('With F(P) = %.3e\n',f(p))
    fprintf('Number of iterations = %3i\n',i)
    fprintf('Tolerance = %.3e |p-pold| = %.3e\n',TOL, err)
end

```

Newton's Method

```

Approximate solution P = 0.00000542
With F(P) = 1.472e-11
Number of iterations = 18
Tolerance = 1.000e-05 |p-pold| = 5.425e-06

```

Published with MATLAB® R2019a