**Problem:**

In this first project we are going to attempt to implement an algorithm which uses Newton’s method (unmodified) to approximate one of the zeros of the function

using an initial approximation of p0 = 0.5. Additional functionality will be added so that the user may specify the tolerance and maximum number of iterations to attempt.

Note that the derivative

has critical values of x = 0 and x = 2. Thus we should expect errant behavior from Newton’s Method at these points. That is, since f’(0) = f’(2) = 0 and

choosing zero or two as an initial approximation would yield a non-existent sequence.

**Source Code:**

% File Name: newton\_hw1.m

% Assignment: Project 1

% Student: Joseph Free

% Course: MATH3261

%

% Purpose: Program calculates the zero of the function

% x^3 - 3\*x^2 + (4/3) within a user specified tolerance through

% an implementation of Newton's method (unmodified).

%

% Required input:

%

% p0 -- The initial approximation of the zero.

% tolerance -- Degree of accuracy.

% maxIter -- Maximum number of iterations to attempt before

% failure.

%

function newton\_hw1 = newton\_hw1( p0, tolerance, maxIter )

% Control variable used to indicate that desired accuracy was achieved.

TOLREACHED = 0;

for i = [1:maxIter]

%Calculate p.

p = p0 - (p0^3-3\*p0^2+4/3)/(3\*p0^2-6\*p0);

if abs(p-p0) < tolerance

p

TOLREACHED = TOLREACHED + 1;

break;

end

% Update approximation.

p0 = p;

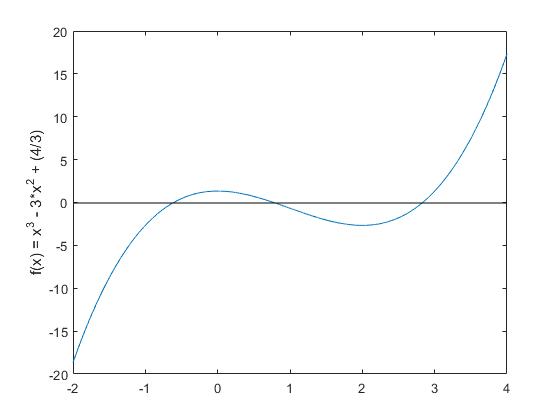
end;

if( TOLREACHED ~= 1 )

fprintf('Method failed after %d iterations\n', maxIter);

end

end

**Sample Output / Solution:**

**Success:** ( For iterations = 10, 5, 3 )

>> newton\_hw1(.5, .001, 10)

p =

0.7739

>> newton\_hw1(.5, .001, 5)

p =

0.7739

>> newton\_hw1(.5, .001, 3)

p =

0.7739

**Figure 1: Graph of the function.**

**Failures:**

>> newton\_hw1(2.0001, .01, 10)

Method failed after 10 iterations

>> newton\_hw1(2, .001, 20)

Method failed after 20 iterations

>> newton\_hw1(1.0e-10, .01, 35)

Method failed after 35 iterations

**Observations:**

From the fundamental theorem of algebra, we know that the function

has at most 3 real roots. From the implemented algorithm, we’ve found one of them to be close to .7739 to within .001 after starting with an initial approximation of .5. From the graph of f, we see this is an agreeable value. Also from the graph, we see the remaining zeros are in (-1,0) and (2, 3), respectively. If we choose instead an initial approximation of 3, we find that Newton’s method yields p = 2.8340 at .001 tolerance in just 3 iterations. Similarly, an initial choice of -.5 with identical parameters yields p = -.6079. Which again seem to agree nicely with the graph of the function.

Note that Newton’s method fails to converge to the zero p = 2.8340 given an initial approximation of 2. This is because the function achieves a local minimum at 2, thereby resulting in an undefined value of pn (because f’(x) appears in the denominator in the iterative method). Similar behavior can be observed with an initial approximation of 0, at which occurs a local maximum.