BISECTION METHOD MATLAB CODE

Numerical Analysis (ENME 602)

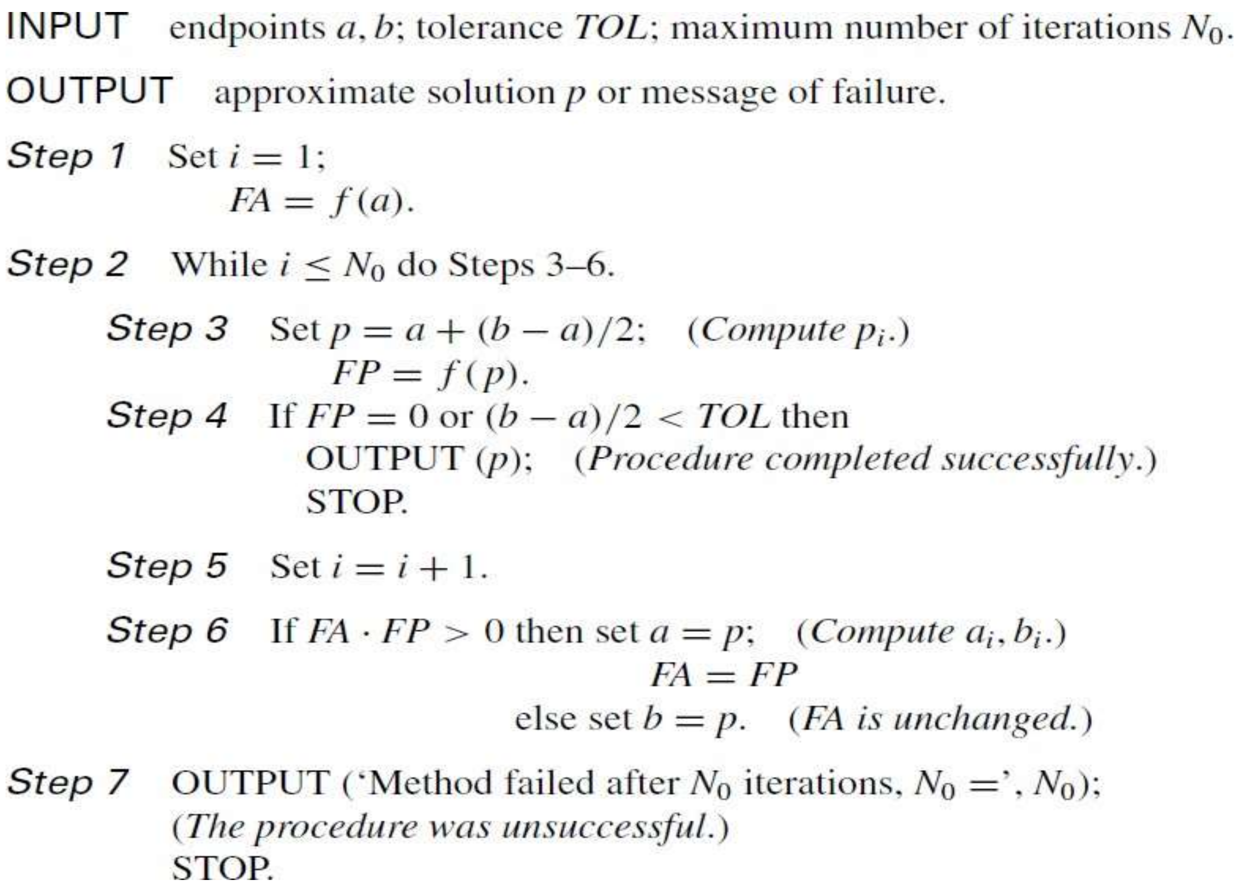
Instructor: Assoc.Prof.Dr. Hesham H. Ibrahim

Eng. Nouran Adel

Outline:

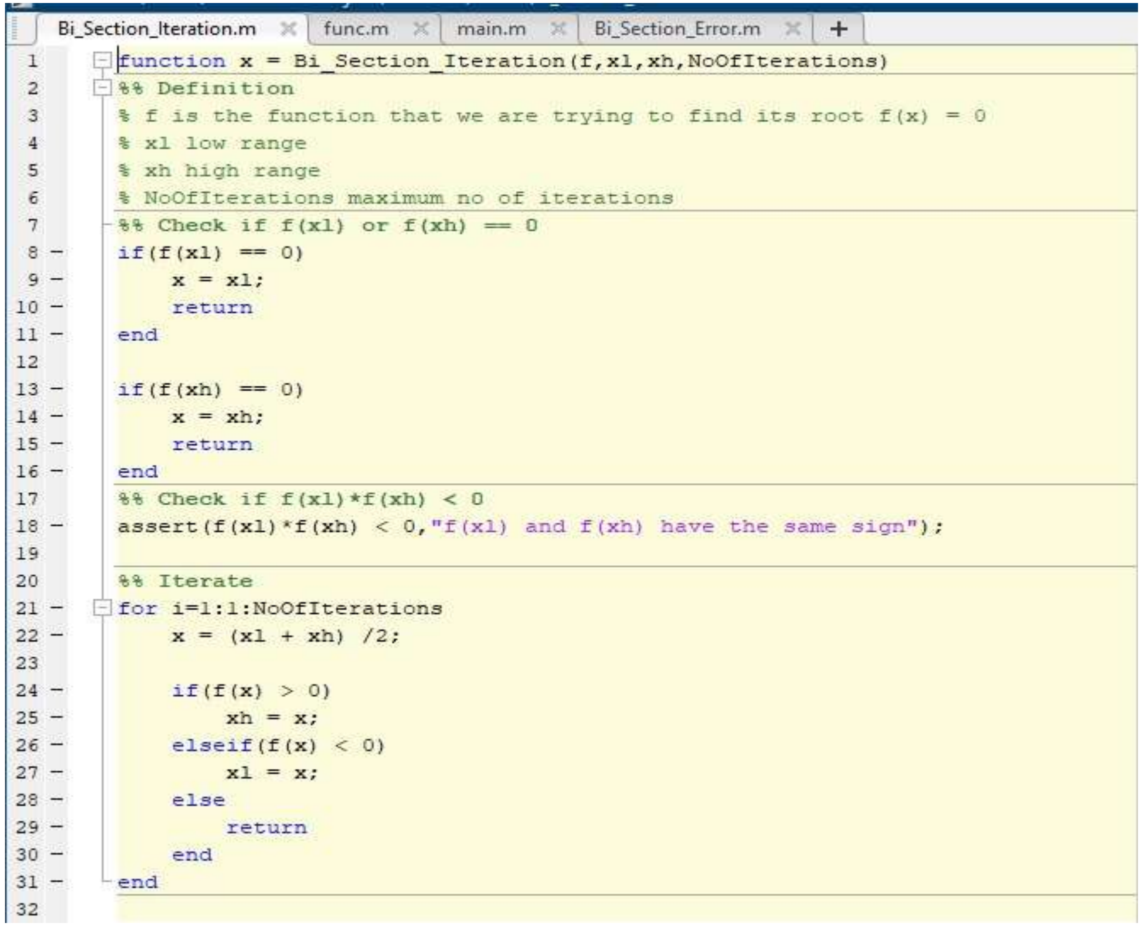
- Pseudo Code.
- Bisection Algorithm.
- Bisection Error.
- Function Algorithm.
- Main Code.

Pseudo Code



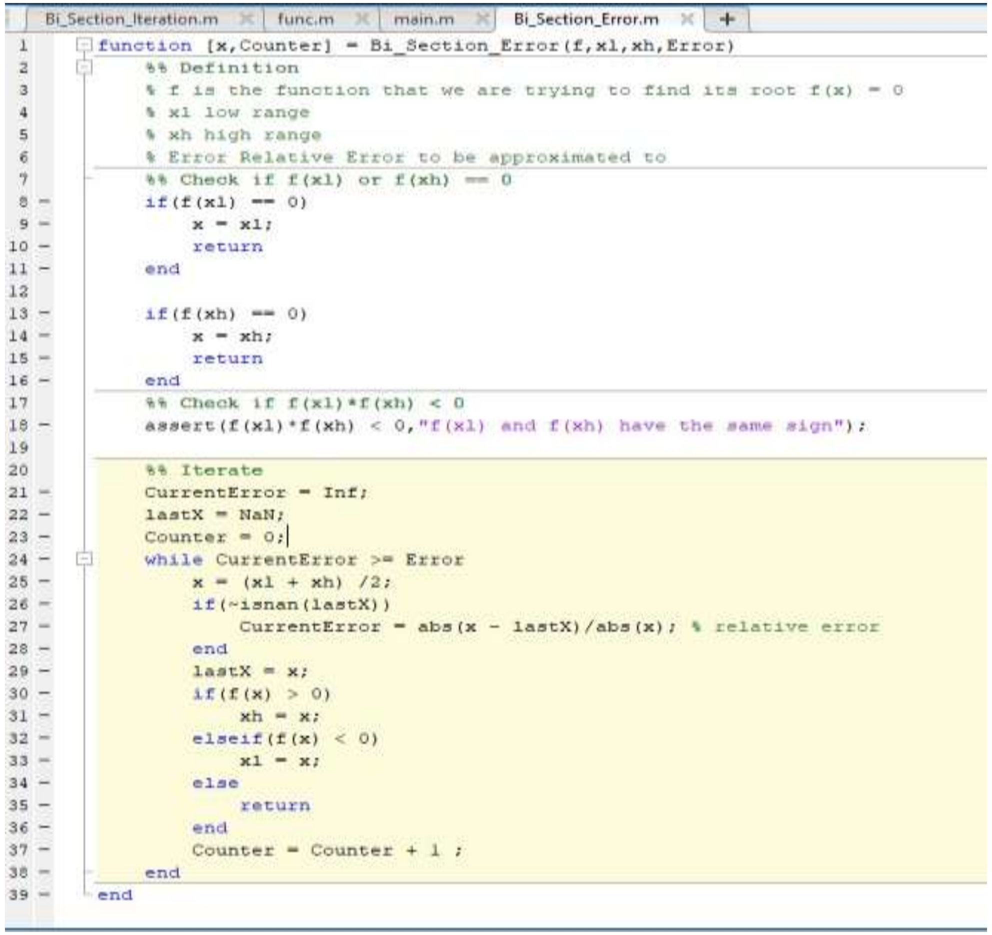
```
INPUT   endpoints $a, b$; tolerance $TOL$; maximum number of iterations $N_0$.
OUTPUT   approximate solution $p$ or message of failure.
Step 1   Set $i = 1$;
             $FA = f(a)$.
Step 2   While $i \leq N_0$ do Steps 3–6.
    Step 3   Set $p = a + (b - a)/2$;   (Compute $p_i$.)
                 $FP = f(p)$.
    Step 4   If $FP = 0$ or $(b - a)/2 < TOL$ then
                 OUTPUT ($p$);   (Procedure completed successfully.)
                 STOP.
    Step 5   Set $i = i + 1$.
    Step 6   If $FA \cdot FP > 0$ then set $a = p$;   (Compute $a_i, b_i$.)
                                          $FA = FP$
                                  else set $b = p$.   ($FA$ is unchanged.)
Step 7   OUTPUT ('Method failed after $N_0$ iterations, $N_0 =$', $N_0$);
         (The procedure was unsuccessful.)
         STOP.
```

Bisection Algorithm



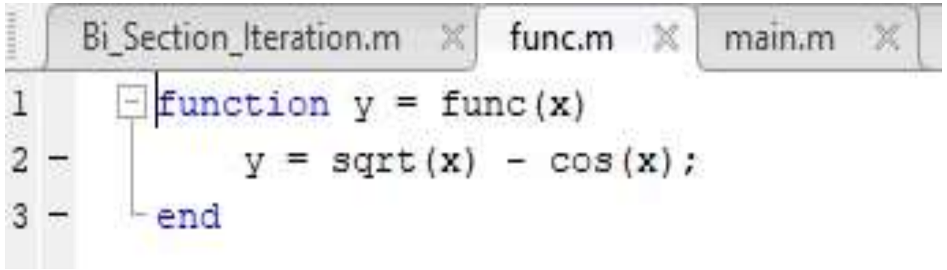
```
function x = Bi_Section_Iteration(f,xl,xh,NoOfIterations)
%% Definition
% f is the function that we are trying to find its root f(x) = 0
% xl low range
% xh high range
% NoOfIterations maximum no of iterations
%% Check if f(xl) or f(xh) == 0
if(f(xl) == 0)
    x = xl;
    return
end

if(f(xh) == 0)
    x = xh;
    return
end
%% Check if f(xl)*f(xh) < 0
assert(f(xl)*f(xh) < 0,"f(xl) and f(xh) have the same sign");

%% Iterate
for i=1:1:NoOfIterations
    x = (xl + xh) /2;

    if(f(x) > 0)
        xh = x;
    elseif(f(x) < 0)
        xl = x;
    else
        return
    end
end

```

Bisection Error



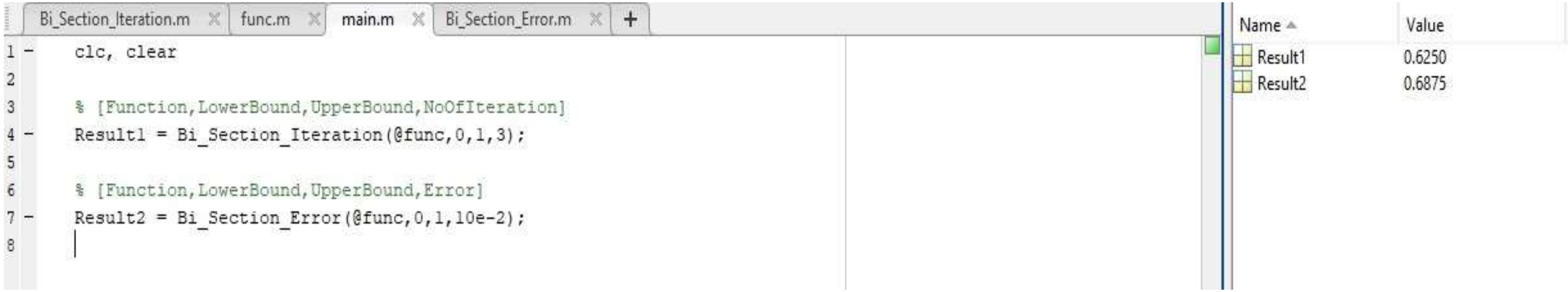
```
function [x,Counter] = Bi_Section_Error(f,xl,xh,Error)
    %% Definition
    % f is the function that we are trying to find its root f(x) = 0
    % xl low range
    % xh high range
    % Error Relative Error to be approximated to
    %% Check if f(xl) or f(xh) == 0
    if(f(xl) == 0)
        x = xl;
        return
    end

    if(f(xh) == 0)
        x = xh;
        return
    end
    %% Check if f(xl)*f(xh) < 0
    assert(f(xl)*f(xh) < 0,"f(xl) and f(xh) have the same sign");

    %% Iterate
    CurrentError = Inf;
    lastX = NaN;
    Counter = 0;
    while CurrentError >= Error
        x = (xl + xh) /2;
        if(~isnan(lastX))
            CurrentError = abs(x - lastX)/abs(x); % relative error
        end
        lastX = x;
        if(f(x) > 0)
            xh = x;
        elseif(f(x) < 0)
            xl = x;
        else
            return
        end
        Counter = Counter + 1 ;
    end
end
```

Function Algorithm



```
function y = func(x)
    y = sqrt(x) - cos(x);
end
```

Main Code



```
clc, clear

% [Function,LowerBound,UpperBound,NoOfIteration]
Result1 = Bi_Section_Iteration(@func,0,1,3);

% [Function,LowerBound,UpperBound,Error]
Result2 = Bi_Section_Error(@func,0,1,10e-2);
```

Name	Value
Result1	0.6250
Result2	0.6875

Thank you!