
Table of Contents

.....	1
bisection function	1
iterator (first pointer)	2
newton function	3
finder function	3

```
% a. f(x) = 2.0 # x + ln(x) = 0
```

```
% function call
```

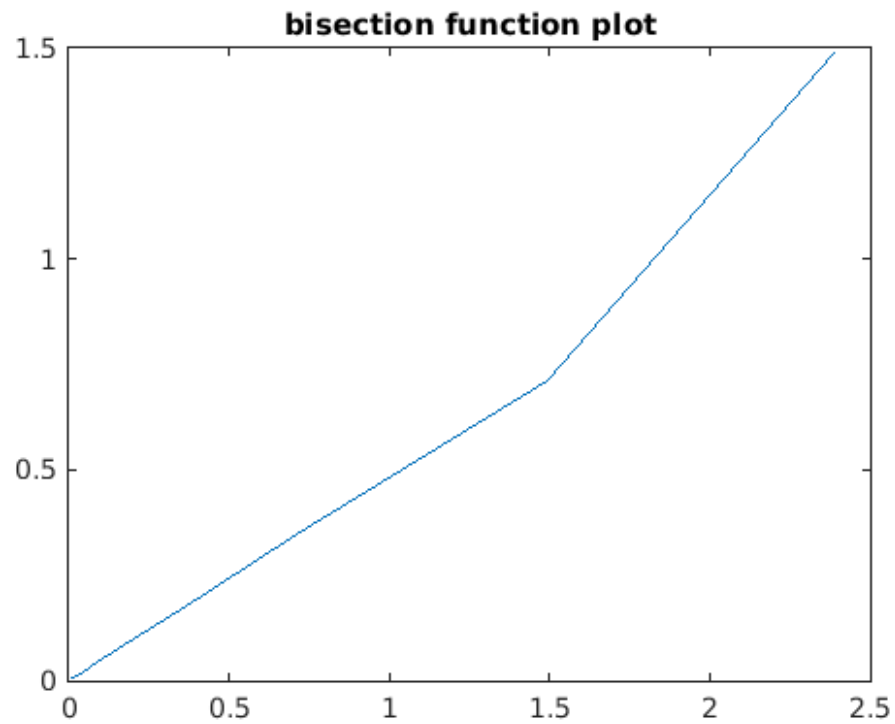
```
bisect()
```

```
fpnr()
```

```
newton()
```

bisection function

```
function bisect()  
    x1 = 1.0;  
    x2 = 5.0;  
    f1 = find(1, x1);  
    f2 = find(1, x2);  
    max_itr = 50;  
    tol = 10^-6;  
    if (f1*f2 > 0)  
        fprintf("INVALID!");  
    end  
    err = zeros(1, max_itr);  
    for itr = 1 : 1 : max_itr  
        m = (x1 + x2)/2;  
        fm = find(1, m);  
        err(itr) = abs(f2 - f1);  
        if (err(itr) < tol)  
            break  
        end  
        if (f1*fm > 0)  
            x1 = m;  
            f1 = fm;  
        else  
            x2 = m;  
            f2 = fm;  
        end  
    end  
    m  
    figure(1)  
    plot(err(1 : itr - 1), err(2 : itr))  
    title('bisection function plot')  
end
```

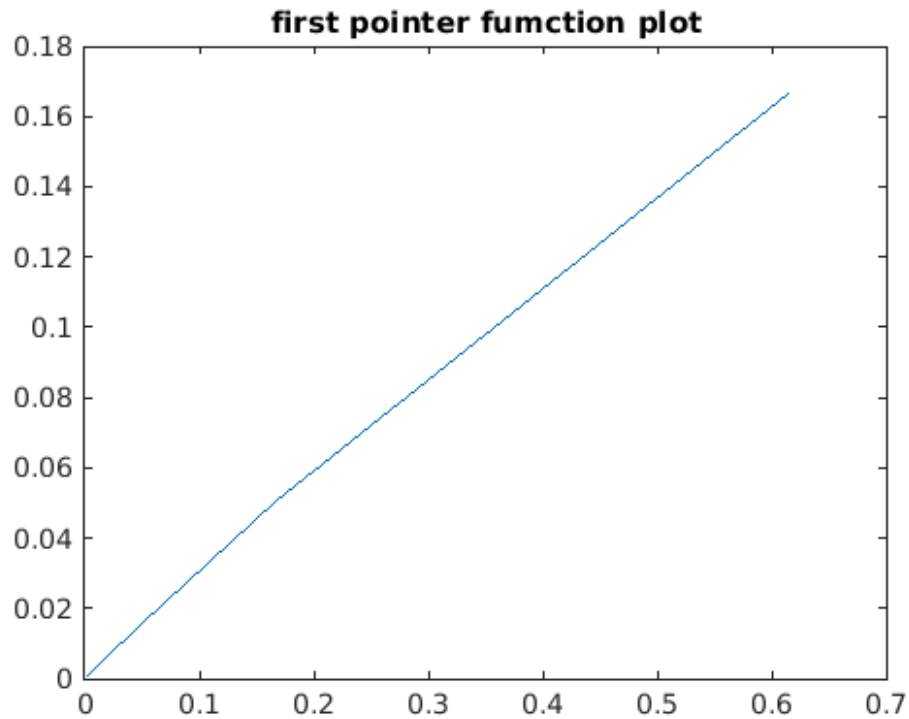


iterator (first pointer)

```
function fpitr()  
    x = 4;  
    xold = 4;  
    max_itr = 50;  
    errfp = zeros(1, max_itr);  
    tol = 10^-6;  
    for itr = 1 : 1 : max_itr  
        x = find(2, xold);  
        errfp(itr) = abs(x - xold);  
        xold = x;  
        if errfp(itr) < tol  
            break  
        end  
    end  
    x  
    figure(2)  
    plot(errfp(1 : itr - 1), errfp(2 : itr))  
    title('first pointer function plot')  
end
```

x =

3.1462



newton function

```
function newton()  
    x = 4;  
    max_itr = 50;  
    errnt = zeros(1, max_itr);  
    tol = 10^-6;  
    for itr = 1 : 1 : max_itr  
        fx = find(1, x);  
        dfx = find(3, x);  
        xnew = x - fx/dfx;  
        errnt(itr) = abs(xnew - x);  
        if(errnt(itr) < tol)  
            break  
        end  
        x = xnew;  
    end  
    xnew  
    figure(3)  
    plot(errnt(1 : itr - 1), errnt(2 : itr))  
    title('newton function plot')  
end
```

finder function

```
function fval = find(arg, x)  
    if arg == 1
```

```
        fval = 2 - x + log(x);  
elseif arg == 2  
    fval = 2 + log(x);  
else  
    fval = 1/(x) - 1;  
end  
end
```

```
m =
```

```
3.1462
```

Published with MATLAB® R2020b