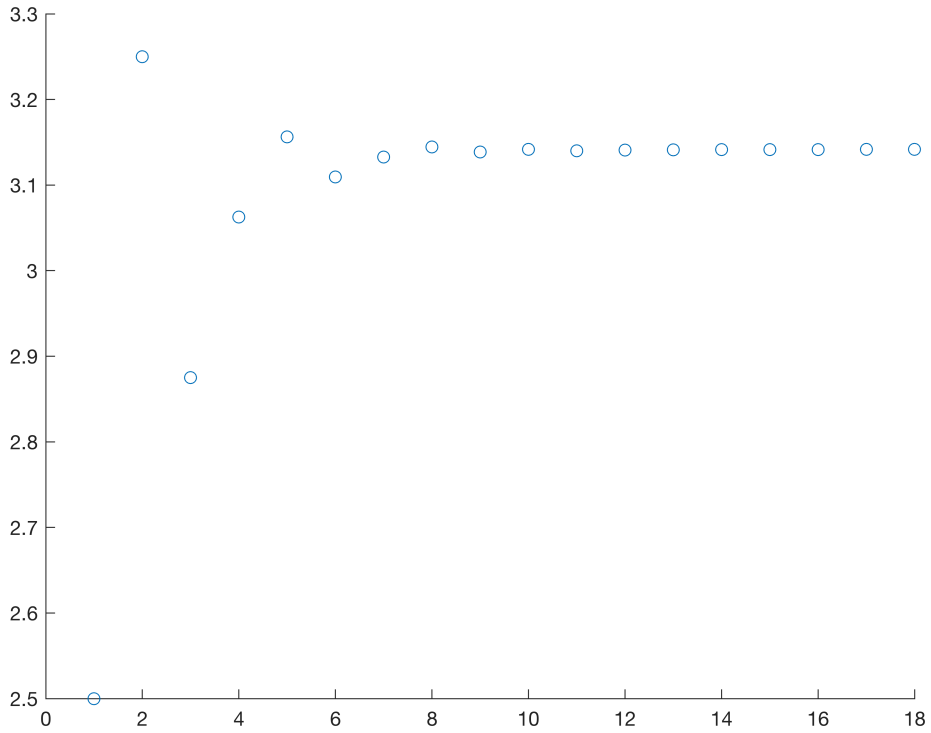
```
%Number 1A
f = @(x) sin(x);
a = 1;
b = 4;
tol = 0.00002;
maxits = 30;

assert(sign(f(a)) ~= sign(f(b)));
assert(a<b);
assert(~isinf(a) && ~isinf(b));

format long
bisection(f,a,b,tol,maxits)
```

```
We are at interation 1
We are at interation 2
We are at interation 3
We are at interation 4
We are at interation 5
We are at interation 6
We are at interation 7
We are at interation 8
We are at interation 9
We are at interation 10
We are at interation 11
We are at interation 12
We are at interation 13
We are at interation 14
We are at interation 15
We are at interation 16
We are at interation 17
We are at interation 18
result =
   3.141590118408203
```




```
ans =
   3.141590118408203
```

```
% 3.14159
% The tolerance has to be relatively low
% tol < 0.00002 to get an accurate reading
% Takes about 17 iterations to get the
% first 6 desired digits
```

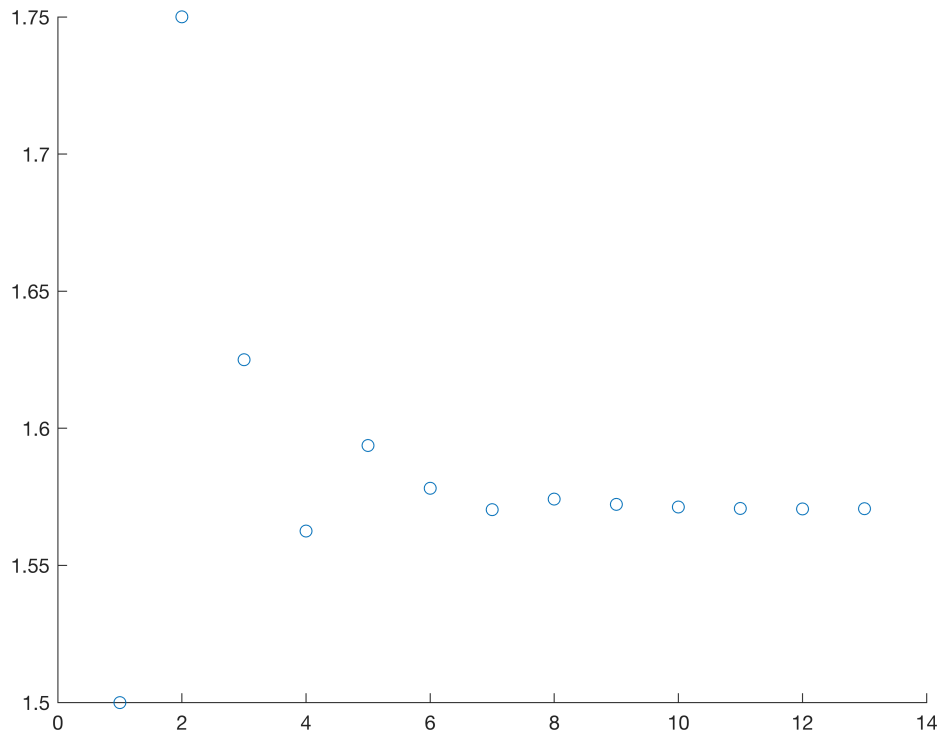
```
%Number 1B
f = @(x) tan(x);
a = 1;
b = 2;
tol = 0.0002;
maxits = 20;

assert(sign(f(a)) ~= sign(f(b)));
assert(a<b);
assert(~isinf(a) && ~isinf(b));

bisection(f,a,b,tol,maxits)
```

```
We are at interation 1
We are at interation 2
We are at interation 3
We are at interation 4
We are at interation 5
We are at interation 6
```

```
We are at interation 7
We are at interation 8
We are at interation 9
We are at interation 10
We are at interation 11
We are at interation 12
We are at interation 13
result =
   1.570678710937500
```




```
ans =
   1.570678710937500
```

```
%1.570
% Similar to number 1, the tol
% has to be fairly low to get
% an accurate reading. 1.570 is
% about the root. Takes about 10
% iterations
```

```
% Number 2A
fzero(@sin,[1,4])
```

```
ans =
   3.141592653589793
```

```
% 3.141592653589793
% This has all the digits correct
```

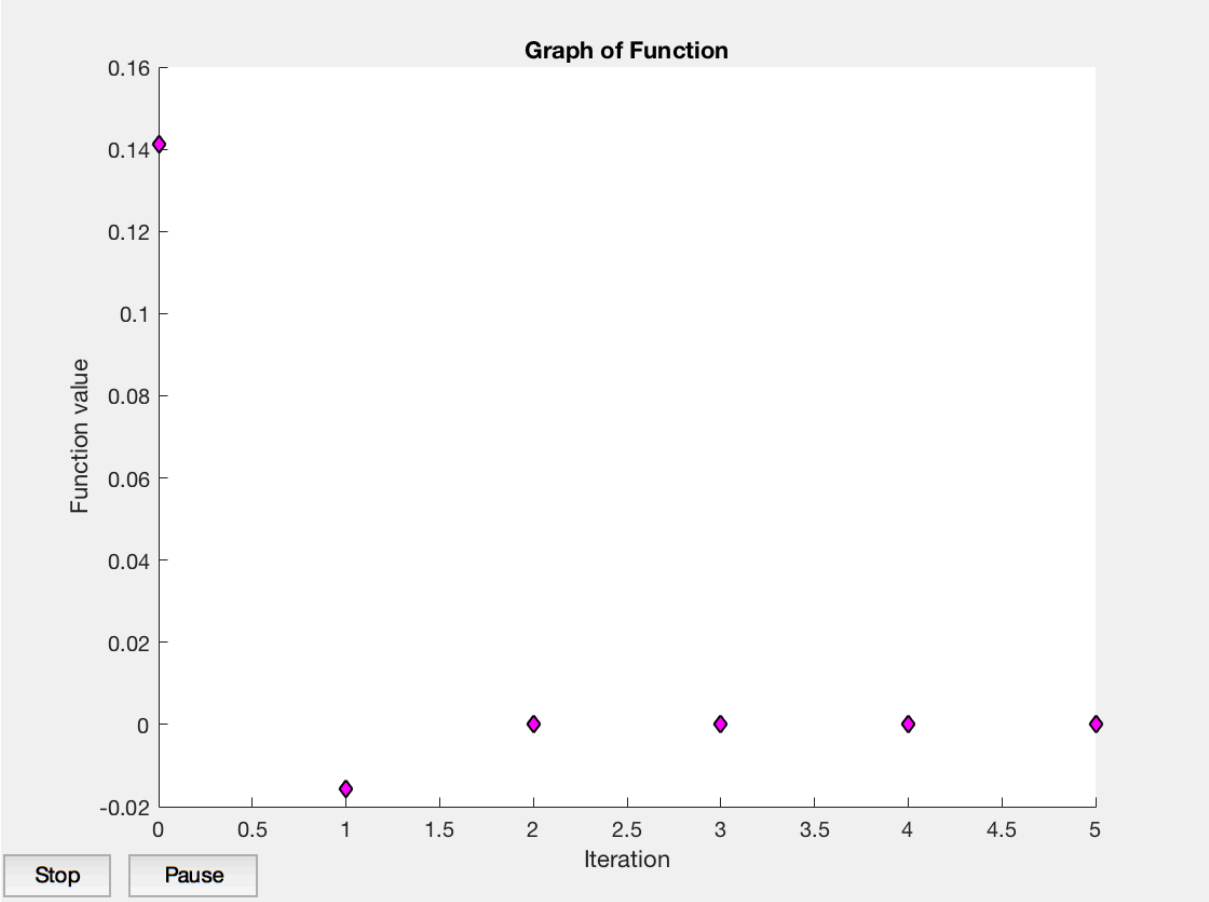
```
%Number 2B
fzero(@tan, [1 2]);
fzero(@tan, [1 2], optimset('Display','iter'));
```

Func-count	x	f(x)	Procedure
2	1	1.55741	initial
3	2	-2.18504	interpolation
4	1.85165	-3.46644	interpolation
5	1.41615	6.4146	bisection
6	1.47895	10.8572	interpolation
7	1.6339	-15.8261	bisection
8	1.61954	-20.4982	interpolation
9	1.58798	-58.1765	bisection
10	1.55642	69.5775	bisection
11	1.55783	77.1347	interpolation
12	1.56502	173.074	bisection
13	1.56861	457.679	bisection
14	1.5722	-710.251	bisection
15	1.57182	-980.91	interpolation
16	1.57041	2574.06	bisection
17	1.57111	-3169.73	interpolation
18	1.57104	-4124.11	interpolation
19	1.57088	-11801.6	bisection
20	1.57072	13697.2	bisection
21	1.57073	14893.3	interpolation
22	1.57077	32636.8	bisection
23	1.57078	80720.7	bisection
24	1.5708	-170547	bisection
25	1.57079	306518	interpolation
26	1.5708	-384460	interpolation
27	1.5708	-515558	interpolation
28	1.5708	1.51194e+06	bisection
29	1.5708	-1.56465e+06	interpolation
30	1.5708	-1.62117e+06	interpolation
31	1.5708	-3.36385e+06	bisection
32	1.5708	-7.27282e+06	bisection
33	1.5708	-1.73587e+07	bisection
34	1.5708	4.48791e+07	bisection
35	1.5708	-5.66155e+07	interpolation
36	1.5708	-7.66641e+07	interpolation
37	1.5708	2.16494e+08	bisection
38	1.5708	-2.37393e+08	interpolation
39	1.5708	-2.62758e+08	interpolation
40	1.5708	-5.88385e+08	bisection
41	1.5708	-1.54689e+09	bisection
42	1.5708	2.45914e+09	bisection
43	1.5708	3.48749e+09	interpolation
44	1.5708	-8.33979e+09	bisection
45	1.5708	1.19881e+10	interpolation
46	1.5708	2.13107e+10	interpolation
47	1.5708	-2.74039e+10	bisection
48	1.5708	-3.1975e+10	interpolation
49	1.5708	-7.67527e+10	bisection
50	1.5708	1.91689e+11	bisection
51	1.5708	-2.56007e+11	interpolation
52	1.5708	-3.85261e+11	interpolation
53	1.5708	7.63028e+11	bisection
54	1.5708	1.49707e+12	interpolation
55	1.5708	-1.55633e+12	bisection
56	1.5708	-1.58761e+12	interpolation
57	1.5708	-3.24064e+12	bisection

```
    58          1.5708  -6.76496e+12        bisection
    59          1.5708  -1.48279e+13        bisection
    60          1.5708  -3.64003e+13        bisection
    61          1.5708   7.86301e+13        bisection
    62          1.5708  -1.33542e+14        interpolation
    63          1.5708   1.93489e+14        interpolation
    64          1.5708   3.66869e+14        interpolation
    65          1.5708  -4.19946e+14        bisection
    66          1.5708  -5.83048e+14        interpolation
    67          1.5708  -1.20927e+15        bisection

Current point x may be near a singular point. The interval [1, 2]
reduced to the requested tolerance and the function changes sign in the interval,
but f(x) increased in magnitude as the interval reduced.
```

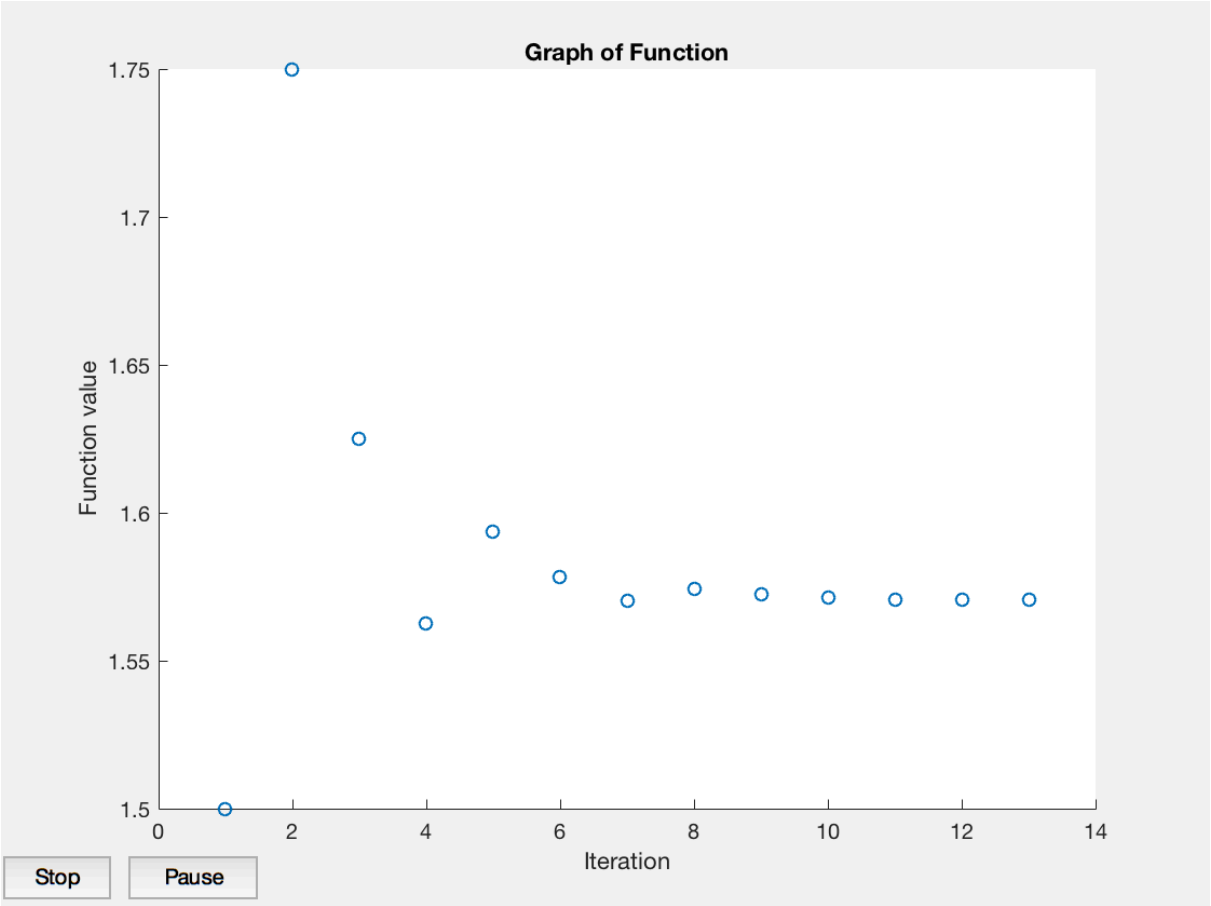
```
%Number 2C
fzero(@sin, [3 4], optimset('PlotFcns',@optimplotfval));
title('Graph of Function');
```




```
%Number 2D
%Update the function
bisection(f,a,b,tol,maxits,'inter');
```

```
We are at interation 1
We are at interation 2
```

```
We are at interation 3
We are at interation 4
We are at interation 5
We are at interation 6
We are at interation 7
We are at interation 8
We are at interation 9
We are at interation 10
We are at interation 11
We are at interation 12
We are at interation 13
result =
   1.570678710937500
```




```
function result = bisection(f, a, b, tol, maxits, iter)
% f is function
% a,b is the interval
% tol is the tolerance before we encounter an error (27)
% maxits is the number of ints allowed

    %fzero(f, [a b]);
    %fzero(f, [a b], optimset('Display','iter'))
    %fzero(f, [a b], optimset('PlotFcns', @optimplotfval))

    if (f(a) == 0)
        result = a;
        return;
```

```
    elseif f(b) == 0
        result = b;
        return;
    end

    x = 0;
    y = 0;

    for i=1:maxits
        fprintf("We are at interation "+ i + "\n");
        c = (a+b)/2;
        x(i) = i;
        y(i) = c;
        if f(c) == 0 || (b-a)/2 < tol
           result = c
           break;
        end
        if sign(f(c)) ~= sign(f(a))
            %print(root if left of the subinterval)
            b = c; %a stays unchanged
        else
            a = c; %b stays unchanged
        end

        result = c;
    end
        if ((b-a)/2 > tol)
            fprintf("We're going to need more iterations!");
        end
        title('Graph of Function');
        scatter(x, y);

end
```