**Bisection Method**

To find the root of

**SCILAB COMMAND**

clc

clear

function **c**=bisection(**f**, **a**, **b**, **epsilon**)

fa = **f**(**a**); fb = **f**(**b**); *// evaluate function at interval endpoints*

**c** = (**a**+**b**)/2; *// get midpoint*

*// check that f changes sign:*

if ( sign(fa) == sign(fb) )

error('error in bisection: f(a), f(b) have the same sign!');

end

*// print header for output table:*

printf(' [ %6s, %6s ] %8s %8s\n','a','b','c','fc')

while ( (**b**-**a**)/2 > **epsilon** ) *// loop til desired accuracy is reached*

fc = **f**(**c**); *// evaluate f at midpoint*

printf(' [ %6.4f, %6.4f ] %8.4f %8.4f\n',**a**,**b**,**c**,fc)

if ( sign ( fc ) == sign ( fa ) ) *// shrink the bracketing interval*

**a** = **c**; fa = fc;

elseif ( sign ( fc ) == sign ( fb ) )

**b** = **c**; fb = fc;

end

**c** = (**a**+**b**)/2; *// get new midpoint*

end *// while*

endfunction

*// define the function whose root we want to find:*

function **y**=myfunc(**x**)

**y** = exp(**x**) - 4\***x**

endfunction

**After running this program type on console**

--> bisection( myfunc, 0, 2, 0.001 )

Then output will be

[ a, b ] c fc

[ 0.0000, 2.0000 ] 1.0000 -1.2817

[ 0.0000, 1.0000 ] 0.5000 -0.3513

[ 0.0000, 0.5000 ] 0.2500 0.2840

[ 0.2500, 0.5000 ] 0.3750 -0.0450

[ 0.2500, 0.3750 ] 0.3125 0.1168

[ 0.3125, 0.3750 ] 0.3438 0.0352

[ 0.3438, 0.3750 ] 0.3594 -0.0051

[ 0.3438, 0.3594 ] 0.3516 0.0150

[ 0.3516, 0.3594 ] 0.3555 0.0050

[ 0.3555, 0.3594 ] 0.3574 -0.0000

ans =

0.3564453