Roots and integrals

Compare the accuracies (how many digits are correct) and efficiencies (how many steps does it take to reach a given accuracy) of any two quadrature algorithms on the definite integral:

$$\int_0^1 \exp(3x)dx$$

And any two root-finding algorithms on the two functions

$$y(x) = \tan(x)$$

and

$$y(x) = \tanh(x)$$

In [1]:

```
import math
import numpy as np
import matplotlib.pyplot as plt
from rootfinding import *
from integrals import *
```

Integrate $\exp(3x)$

True value is

$$\int_0^1 \exp(3x)dx = \frac{1}{3}(\exp(3) - 1) = 6.361845641062556$$

Plot the function

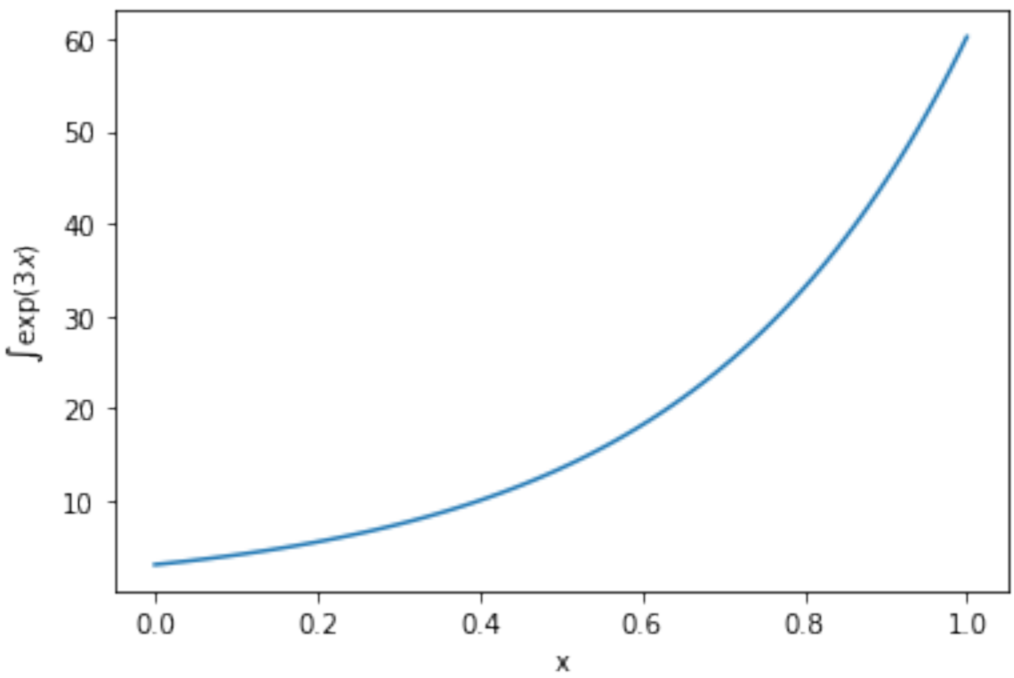
In [2]:

```
x0 = 0
x1 = 1
n = 101
xvals = np.linspace(x0,x1,n)
fvals = np.exp( 3* xvals )
ana_der = 3 * fvals
```

In [3]:

```
plt.plot(xvals,ana_der)
plt.xlabel("x")
plt.ylabel("$\int \exp(3x)$")

I = (np.exp(3) - 1 ) / 3.
print("True value of integral = %6.4f" % (I))
```

```
True value of integral = 6.3618
```



Compute the integrals

In [4]:

```
I1 = simpson(lambda x : np.exp(3*x), x0, x1, n)
I2 = trapezoid(lambda x : np.exp(3*x), x0, x1, n)
I3 = adaptive_trapezoid(lambda x : np.exp(3*x), x0, x1, 1e-6, ou
tput=True)
```

```
N = 2,  Integral = 10.542768461593834
N = 2.0,  Integral = 7.512228765965949
N = 4.0,  Integral = 6.657298346225774
N = 8.0,  Integral = 6.436224369354567
N = 16.0,  Integral = 6.380472949015574
N = 32.0,  Integral = 6.366504513566256
N = 64.0,  Integral = 6.363010487133510 5
N = 128.0,  Integral = 6.362136860578428 5
N = 256.0,  Integral = 6.361918446441433
N = 512.0,  Integral = 6.361863842438524
N = 1024.0,  Integral = 6.3618501914085
N = 2048.0,  Integral = 6.361846778649171
N = 4096.0,  Integral = 6.361845925459217
```

Compare accuracies

In [5]:

```
print("Accuracies: ")
print( "True integral       = %6.6e" % (I) )
print( "Simpson's rule      = %6.6e, err = %6.6e " % ( I1, (I1 -
I)/I )  )
print( "Trapezoid rule      = %6.6e, err = %6.6e " % ( I2, (I2 -
I)/I )  )
print( "Adaptive trap rule = %6.6e, err = %6.6e " % ( I2, (I2 -
I)/I )  )
```

```
Accuracies:
True integral      = 6.361846e+00
Simpson's rule     = 6.167843e+00, err = -3.049464e-
02
Trapezoid rule     = 6.362313e+00, err = 7.352112e-0
5
Adaptive trap rule = 6.362313e+00, err = 7.352112e-0
5
```

Find the roots of $\tan(x)$ and $\tanh(x)$

Be careful! Both of these have roots at $x = 0$, so for the purposes of the problem, choose something else!

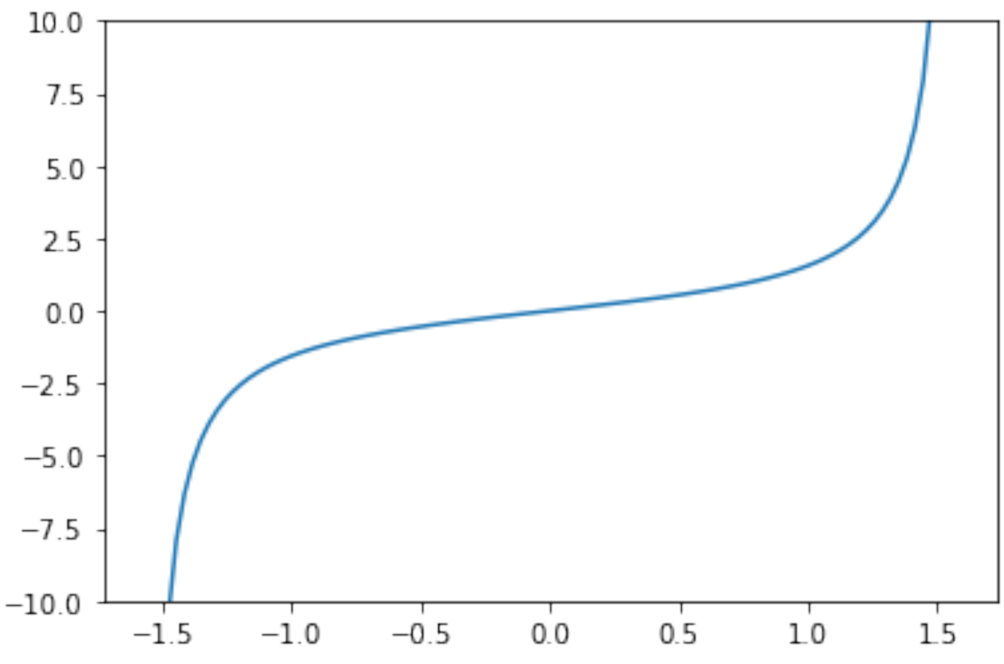
Plot the functions

In [6]:

```
x0 = -np.pi*0.5
x1 = np.pi*0.5
xvals1 = np.linspace(x0,x1,n)
f1 = np.tan(xvals1)
plt.plot(xvals1,f1)
plt.ylim(-10,10)
```

Out[6]:

```
(-10, 10)
```

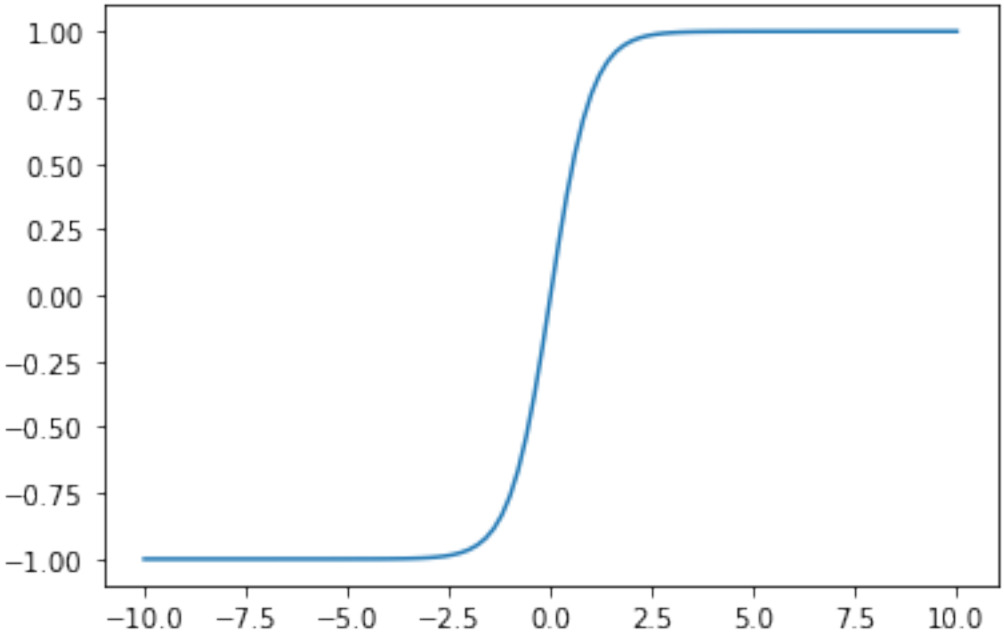


In [7]:

```
x0 = -10
x1 = 10
xvals2 = np.linspace(x0,x1,n)
f2 = np.tanh(xvals2)
plt.plot(xvals2, f2)
```

Out[7]:

```
[<matplotlib.lines.Line2D at 0x7f4b5e906210>]
```



Compute the roots, plot the convergences

In [8]:

```
for i,name,func,deriv in zip(
    range(2),
    ['tan', 'tanh'],
    [lambda x : np.tan(x), lambda x : np.tanh(x)],
    [lambda x : 1./(np.cos(x))**2, lambda x : 1-np.tanh(x)**2 ]
):

    fig = plt.figure(i)
    xroot_simple, iters_simple = root_simple( func, 1.0, -0.1, r
oot_debug=True)
    iternum_simple = np.arange(len(iters_simple))

    xroot_bisection, iters_bisection = root_bisection( func, -0.
5, 1.0, root_debug=True)
    iternum_bisection = np.arange(len(iters_bisection))

    xroot_secant, iters_secant = root_secant( func, -0.5, 1.0, r
oot_debug=True)
    iternum_secant = np.arange(len(iters_secant))

    xroot_tangent, iters_tangent = root_tangent( func, deriv, 1.
0, root_debug=True)
    iternum_tangent = np.arange(len(iters_tangent))


    plt.plot(iternum_simple,iters_simple[:,1],       label='Simp
le      : %6.3e' % (np.abs(xroot_simple)))
    plt.plot(iternum_bisection,iters_bisection[:,1], label='Bise
ction : %6.3e' % (np.abs(xroot_bisection)))
    plt.plot(iternum_secant,iters_secant[:,1],       label='Seca
nt      : %6.3e' % (np.abs(xroot_secant)))
    plt.plot(iternum_tangent,iters_tangent[:,1],     label='Tang
ent     : %6.3e' % (np.abs(xroot_tangent)))
    plt.title(name)
    plt.legend()
```

```
 ROOT FINDING using Simple Search with Step Halving
 Requested accuracy = 1e-06
 Step     Guess For Root          Step Size
Function Value
 ----  --------------------  --------------------
```

```
--------------------
    0  1.0                  -0.1
1.5574077246549023
    1  0.9                  -0.1
1.2601582175503392
    2  0.8                  -0.1
1.0296385570503641
    3  0.7000000000000001    -0.1
0.8422883804630795
    4  0.6000000000000001    -0.1
0.6841368083416924
    5  0.5000000000000001    -0.1
0.5463024898437907
    6  0.40000000000000013   -0.1
0.42279321873816195
    7  0.30000000000000016   -0.1
0.3093362496096234
    8  0.20000000000000015   -0.1
0.20271003550867264
    9  0.10000000000000014   -0.1
0.10033467208545069
   10  1.3877787807814457e-16  -0.1
1.3877787807814457e-16
   11  1.3877787807814457e-16  -0.05
-0.1003346720854541
   12  1.3877787807814457e-16  -0.025
-0.05004170837553865
   13  1.3877787807814457e-16  -0.0125
-0.02500520963574601
   14  1.3877787807814457e-16  -0.00625
-0.012500651082359206
   15  1.3877787807814457e-16  -0.003125
-0.006250081381479781
   16  1.3877787807814457e-16  -0.0015625
-0.003125010172 56564
   17  1.3877787807814457e-16  -0.00078125
-0.0015625012715668582
   18  1.3877787807814457e-16  -0.000390625
-0.0007812501589456195
   19  1.3877787807814457e-16  -0.0001953125
-0.00039062501986807737
   20  1.3877787807814457e-16  -9.765625e-05
-0.00019531250248338813
   21  1.3877787807814457e-16  -4.8828125e-05
-9.765625031030208e-05
```

```
   22  1.3877787807814457e-16  -2.44140625e-05
-4.882812503866633e-05
   23  1.3877787807814457e-16  -1.220703125e-05
-2.441406250471863e-05
   24  1.3877787807814457e-16  -6.103515625e-06
-1.220703125046755 3e-05
   25  1.3877787807814457e-16  -3.0517578125e-06
-6.103515624937013e-06
   26  1.3877787807814457e-16  -1.52587890625e-06
-3.051757812370696 4e-06
   27  1.3877787807814457e-16  -7.62939453125e-07
-1.525878906112406 4e-06

 ROOT FINDING using Bisection Search
 Requested accuracy = 1e-06
 Step     Guess For Root           Step Size
Function Value
 ----  --------------------  --------------------
--------------------
    0  0.25                  1.5
0.2553419212210362 7
    1  -0.125                0.75
-0.1256551365751309 7
    2  0.0625                0.375
0.0625815075662750 2
    3  -0.03125              0.1875
-0.0312601765012559 54
    4  0.015625              0.09375
0.0156262716899438 25
    5  -0.0078125            0.046875
-0.0078126589496000 08
    6  0.00390625            0.0234375
0.0039062698683361 916
    7  -0.001953125          0.01171875
-0.0019531274835306 55
    8  0.0009765625          0.005859375
0.0009765628104409 765
    9  -0.00048828125        0.0029296875
-0.0004882812888051 109
   10  0.000244140625        0.00146484375
0.0002441406298506 385
   11  -0.0001220703125      0.000732421875
-0.0001220703131063 298
   12  6.103515625e-05       0.0003662109375
6.103515632579122e-05
```

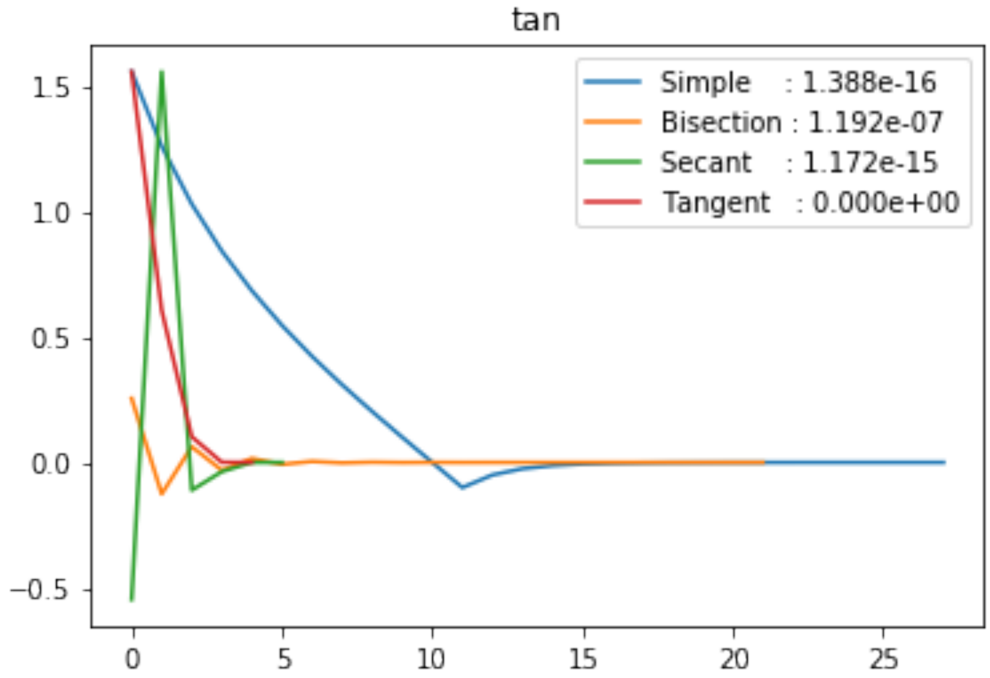
```
   13  -3.0517578125e-05     0.00018310546875
-3.05175781344739e-05
   14  1.52587890625e-05     9.1552734375e-05
1.52587890636842 37e-05
   15  -7.62939453125e-06    4.57763671875e-05
-7.62939453139803e-06
   16  3.814697265625e-06    2.288818359375e-05
3.81469726564350 34e-06
   17  -1.9073486328125e-06  1.1444091796875e-05
-1.90734863281481 3e-06
   18  9.5367431640625e-07   5.7220458984375e-06
9.53674316406539e-07
   19  -4.76837158203125e-07  2.86102294921875e-06
-4.76837158203161 1e-07
   20  2.384185791015625e-07  1.430511474609375e-06
2.38418579101567e-07
   21  -1.1920928955078125e-07  7.152557373046875e-
07  -1.19209289550781 8e-07

 ROOT FINDING using Secant Search
 Requested accuracy = 1e-06
 Step     Guess For Root          Step Size
Function Value
 ----  --------------------  --------------------
--------------------
    0  -0.5                  1.5
-0.5463024898437905
    1  -0.110472141496466    -1.110472141496466
1.557407724654902 3
    2  -0.0366392513066756 4  0.073832890189790 36
-0.1109237503502161 4
    3  -0.000198394974726602 56  0.036440856331949 04
-0.036655655383181 67
    4  -8.926592224366256e-08  0.000198305708804358
9  -0.000198394977329582 1
    5  -1.171712706451702 6e-15  8.926592107194986e-
08  -8.9265922243662 8e-08

 ROOT FINDING using Tangent Search
 Requested accuracy = 1e-06
 Step     Guess For Root          Step Size
Function Value
 ----  --------------------  --------------------
--------------------
    0  1.0                   -0.4546487134128409 7
```

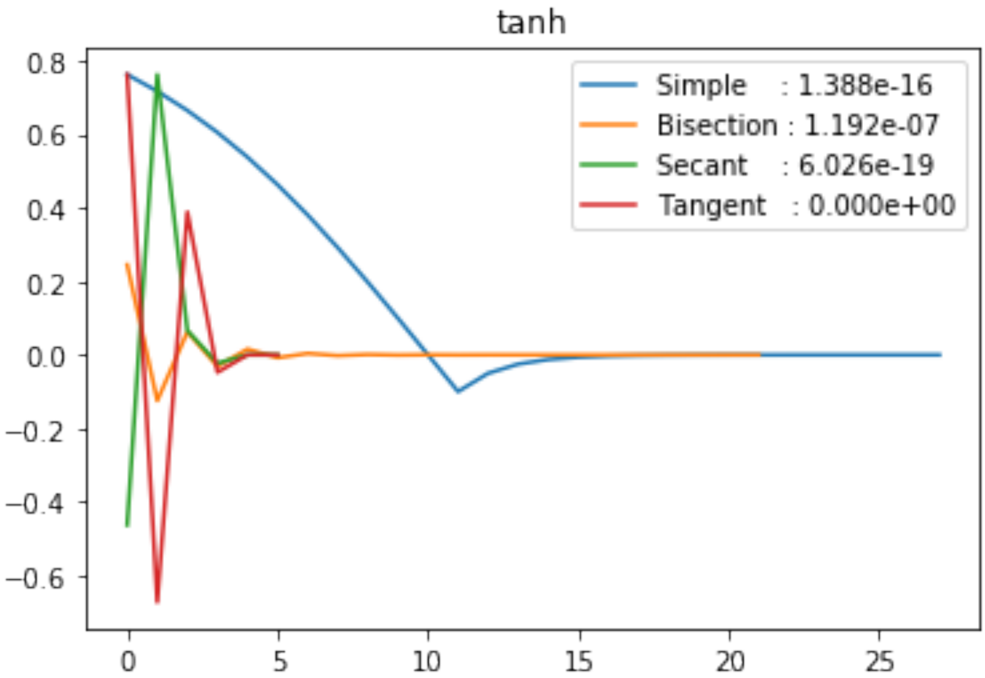
```
1.5574077246549023
    1  0.545351286587159     -0.45464871341284097
0.6067271901700187
    2  0.10187547388140661   -0.4434758127057524
0.10222938520872517
    3  0.00070342232244294924  -0.10117205155697712
0.00070342244044799966
    4  2.3203694564724597e-10  -0.000703422092392454
67  2.3203694564724597e-10

 ROOT FINDING using Simple Search with Step Halving
 Requested accuracy = 1e-06
 Step     Guess For Root           Step Size
Function Value
 ----  --------------------  --------------------
--------------------
    0  1.0                   -0.1
0.7615941559557649
    1  0.9                   -0.1
0.7162978701990245
    2  0.8                   -0.1
0.6640367702678491
    3  0.7000000000000001    -0.1
0.6043677771171636
    4  0.6000000000000001    -0.1
0.5370495669980354
    5  0.5000000000000001    -0.1
0.46211715726000985
    6  0.40000000000000013   -0.1
0.379948962255225
    7  0.30000000000000016   -0.1
0.291312612451591
    8  0.20000000000000015   -0.1
0.19737532022490414
    9  0.10000000000000014   -0.1
0.09966799462495596
   10  1.3877787807814457e-16  -0.1
1.3877787807814457e-16
   11  1.3877787807814457e-16  -0.05
-0.09966799462495568
   12  1.3877787807814457e-16  -0.025
-0.04995837495787984
   13  1.3877787807814457e-16  -0.0125
-0.02499479296842055
   14  1.3877787807814457e-16  -0.00625
```

```
-0.012499348999020728
   15  1.3877787807814457e-16  -0.003125
-0.0062499186210630735
   16  1.3877787807814457e-16  -0.0015625
-0.003124989827513556
   17  1.3877787807814457e-16  -0.00078125
-0.001562498728435348
   18  1.3877787807814457e-16  -0.000390625
-0.0007812498410541806
   19  1.3877787807814457e-16  -0.0001953125
-0.0003906249801316475
   20  1.3877787807814457e-16  -9.765625e-05
-0.0001953124975163344
   21  1.3877787807814457e-16  -4.8828125e-05
-9.765624968942037e-05
   22  1.3877787807814457e-16  -2.44140625e-05
-4.882812496105612e-05
   23  1.3877787807814457e-16  -1.220703125e-05
-2.4414062495010584e-05
   24  1.3877787807814457e-16  -6.103515625e-06
-1.2207031249254893e-05
   25  1.3877787807814457e-16  -3.0517578125e-06
-6.103515624785431e-06
   26  1.3877787807814457e-16  -1.52587890625e-06
-3.0517578123517487e-06
   27  1.3877787807814457e-16  -7.62939453125e-07
-1.5258789061100378e-06

 ROOT FINDING using Bisection Search
 Requested accuracy = 1e-06
 Step     Guess For Root          Step Size
Function Value
 ----  --------------------  --------------------
--------------------
    0  0.25                  1.5
0.24491866240370913
    1  -0.125                0.75
-0.1243530017715962
    2  0.0625                0.375
0.062418746747512514
    3  -0.03125              0.1875
-0.031239831446031256
    4  0.015625              0.09375
0.015623728558408866
    5  -0.0078125            0.046875
```

```
-0.007812341058161014
    6  0.00390625           0.0234375
0.00390623013190634
    7  -0.001953125         0.01171875
-0.001953122516476924
    8  0.0009765625         0.005859375
0.0009765621895592603
    9  -0.00048828125       0.0029296875
-0.0004882812111948964
   10  0.000244140625       0.00146484375
0.00024414062014936172
   11  -0.0001220703125     0.000732421875
-0.0001220703118936702
   12  6.103515625e-05      0.0003662109375
6.103515617420877e-05
   13  -3.0517578125e-05    0.00018310546875
-3.05175781155261e-05
   14  1.52587890625e-05    9.1552734375e-05
1.5258789061315762e-05
   15  -7.62939453125e-06   4.57763671875e-05
-7.629394531101971e-06
   16  3.814697265625e-06   2.288818359375e-05
3.814697265606496e-06
   17  -1.9073486328125e-06  1.1444091796875e-05
-1.907348632810187e-06
   18  9.5367431640625e-07   5.7220458984375e-06
9.536743164059608e-07
   19  -4.76837158203125e-07  2.86102294921875e-06
-4.768371582030884e-07
   20  2.384185791015625e-07  1.430511474609375e-06
2.3841857910155797e-07
   21  -1.1920928955078125e-07  7.152557373046875e-
07  -1.1920928955078068e-07

 ROOT FINDING using Secant Search
 Requested accuracy = 1e-06
 Step     Guess For Root          Step Size
Function Value
 ----  --------------------  --------------------
--------------------
    0  -0.5                 1.5
-0.46211715726000974
    1  0.06645364670890175   -0.9335463532910983
0.7615941559557649
    2  -0.02264733074640214  -0.08910097745530389
```

```
0.06635599749757087
    3  2.19579038740203e-05  0.022669288650276162
-0.022643459589959 9
    4  -3.750312613512777e-09  -2.196165418663381 4e
-05  2.195790387049130 5e-05
    5  6.026341634693828e-19  3.750312614115411e-09
-3.750312613512777e-09

 ROOT FINDING using Tangent Search
 Requested accuracy = 1e-06
 Step     Guess For Root           Step Size
Function Value
 ----  --------------------  --------------------
--------------------
    0  1.0                   -1.81343020392350 9
0.7615941559557649
    1  -0.813430203923509 1   -1.81343020392350 9
-0.671478184137395 7
    2  0.409402316583853 3   1.222832520506894 4
0.387965076018474 74
    3  -0.047304916455615 41  -0.456707233039000 74
-0.047269662404905 18
    4  7.060280364457744e-05  0.047375519259259986
7.060280352726485e-05
    5  -2.34625158385994e-13  -7.06028038792026e-05
-2.34625158385994e-13
```




tanh
Simple : 1.388e-16
Bisection : 1.192e-07
Secant : 6.026e-19
Tangent : 0.000e+00
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
0
5
10
15
20
25

In [ ]: