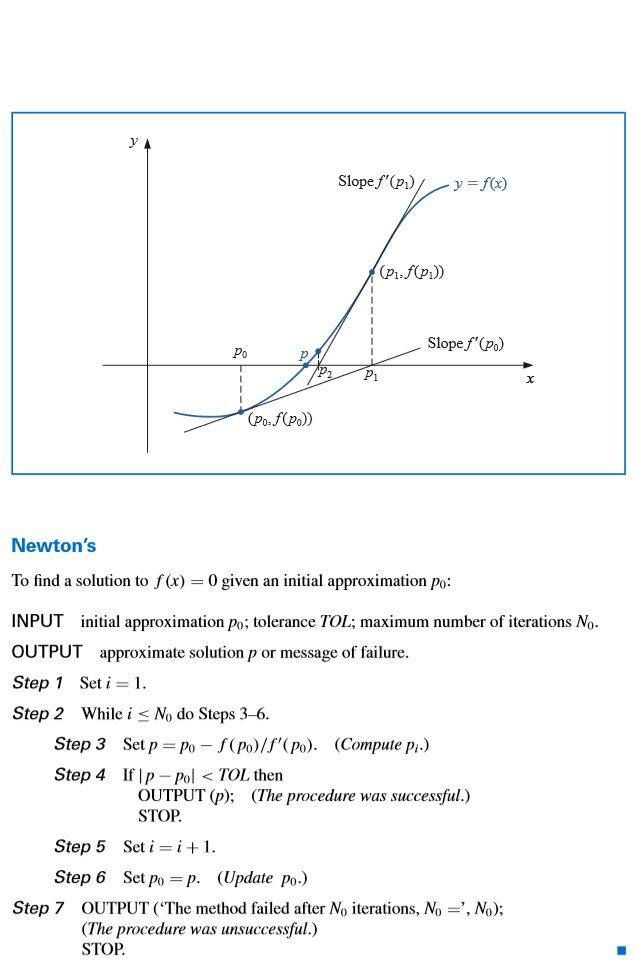
***Newton’s Method – Finding root***



**Newton Method code in Python:**

import numpy as np  
from sympy import \*  
import math  
  
  
def newton\_raphson(f, x0, k):  
 *"""* ***:param*** *f: function.* ***:param*** *x0: start point.* ***:param*** *k: approx Size.* ***:return****: Approximately root value.  
 """* dfunc = lambdify(x, f.diff(x))  
 func = lambdify(x, f)  
 current\_x = x0  
 for i in range(0, k):  
 print(current\_x)  
 current\_x = current\_x - func(current\_x) / dfunc(current\_x)  
 y = lambdify(x, current\_x)  
 return y(x0)  
  
  
e = 2.718281828  
x = Symbol('x')  
y = x \* e \*\* -x - 0.25  
print("{0:.10f}".format(newton\_raphson(y, 2, 10)))

**Function example : xe^-x -0.25**

**F – function**

**X0 – first guess**

**K – number of iterations**

**Results of the code : 2.152735975942831**

**2.153292344259481**

**2.153292364789284**

**2.153292364789284**

**2.153292364789284**

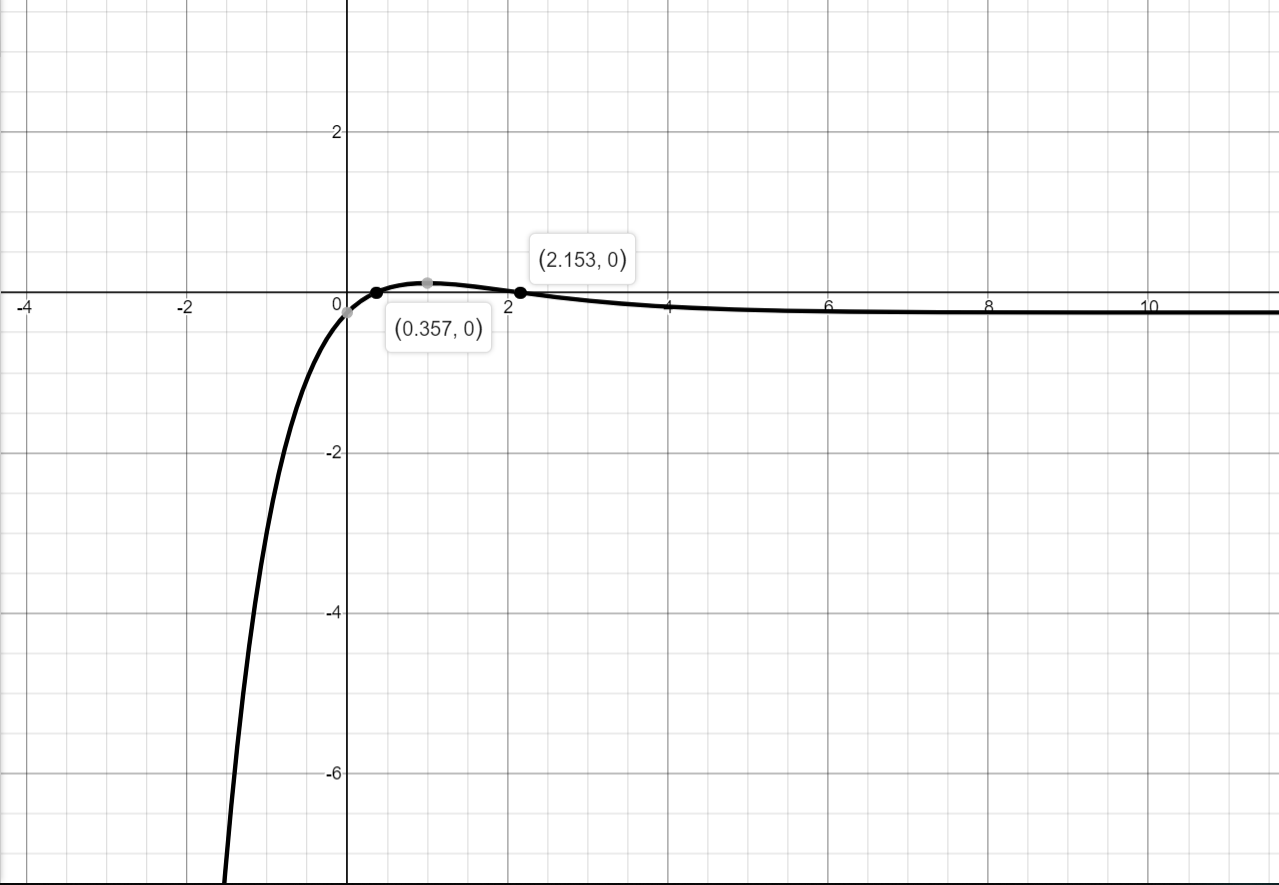
**2.153292364789284**

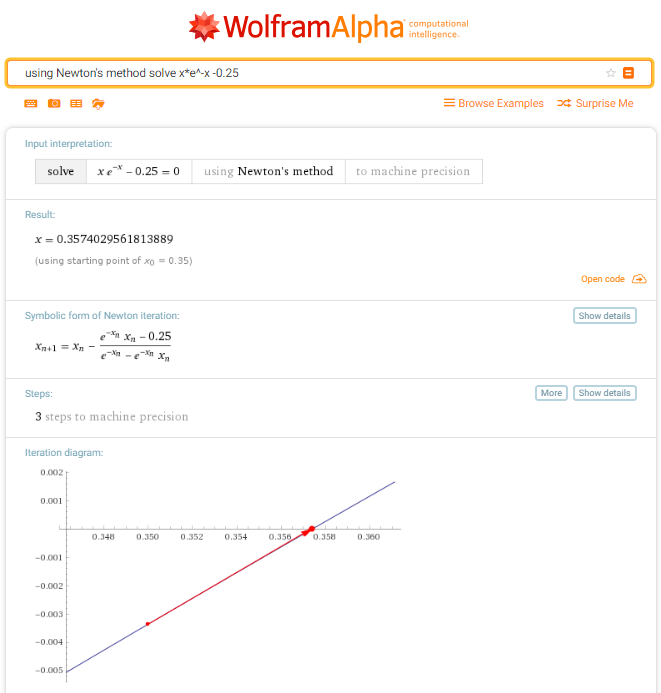
**2.153292364789284**

**2.153292364789284**

**2.153292364789284**

**2.1532923648**

****

****