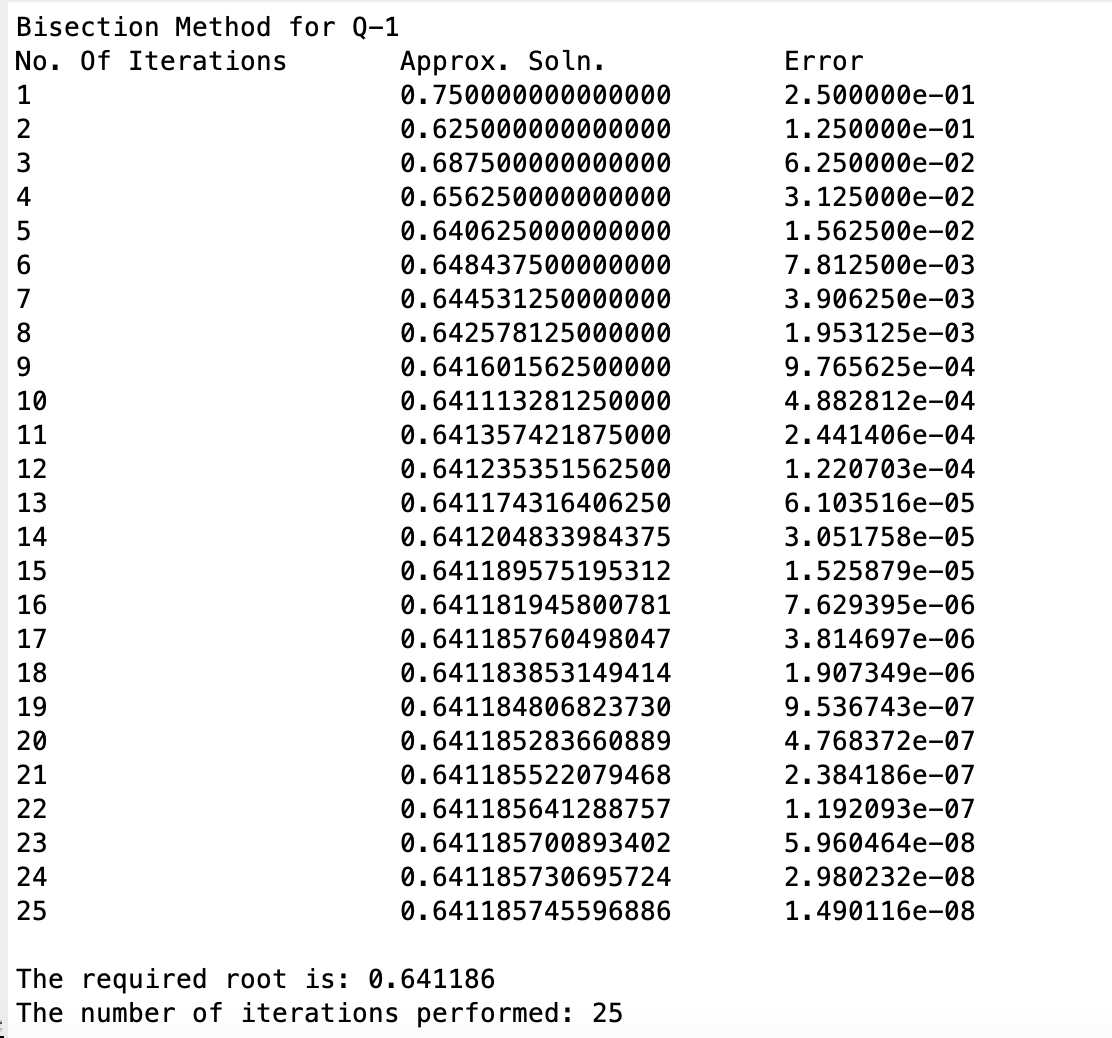
MA322 – Scientific Computing Lab

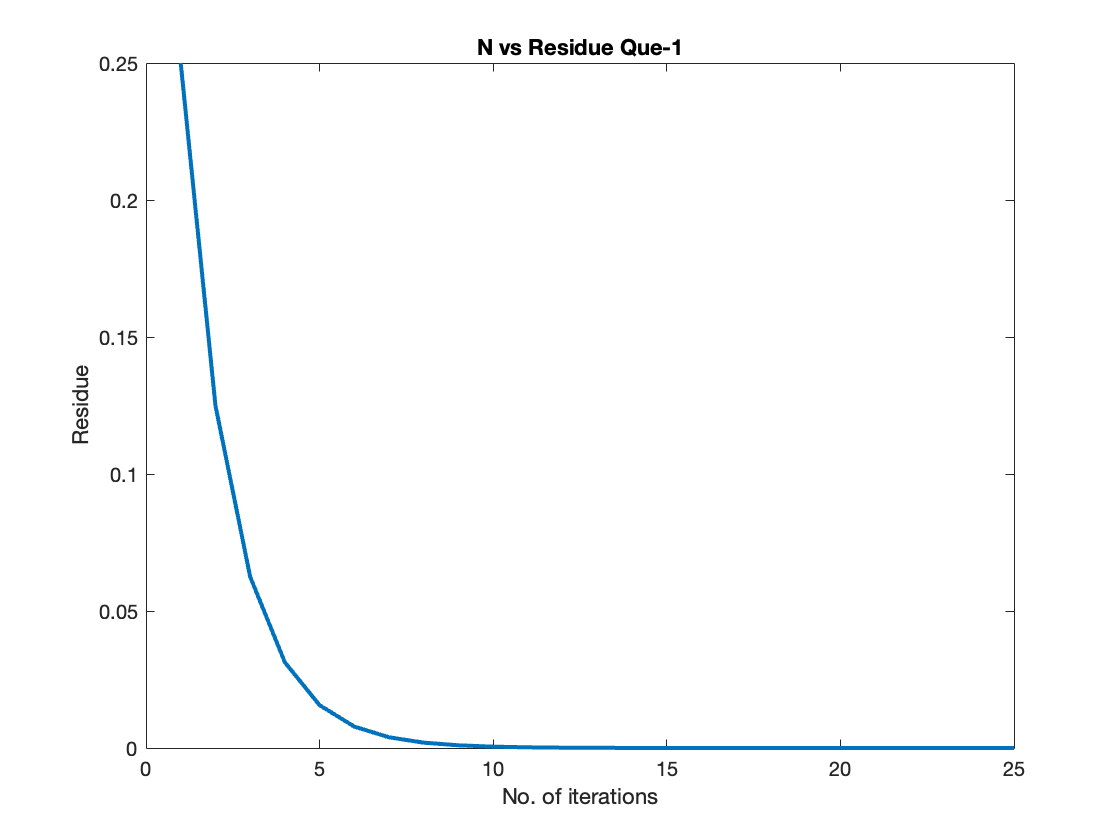
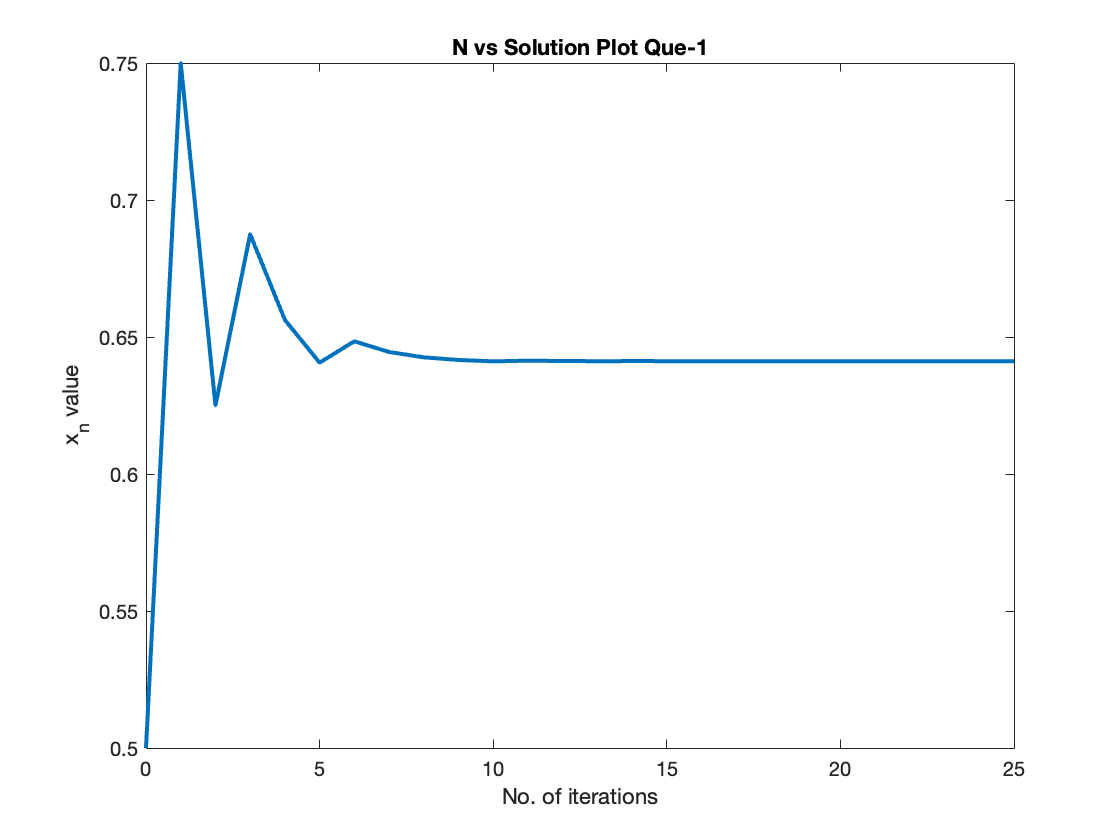
Lab - 02

Dipanshu Goyal 210123083

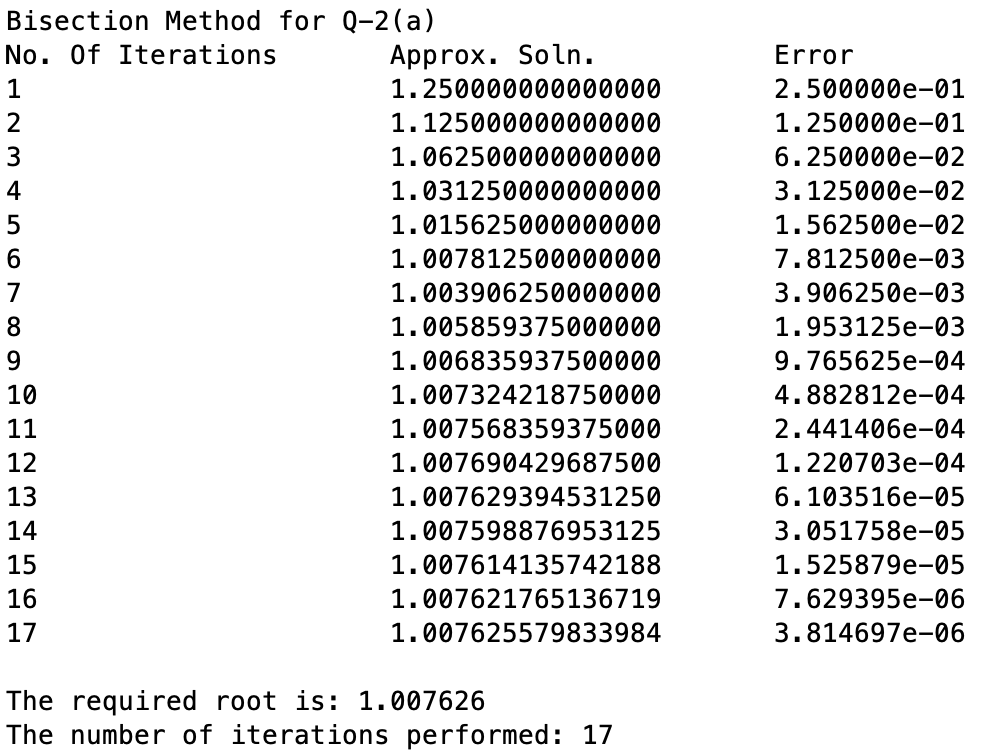
# The residue or error is taken as |xn+1 - xn|. Run the file named output\_file.m.

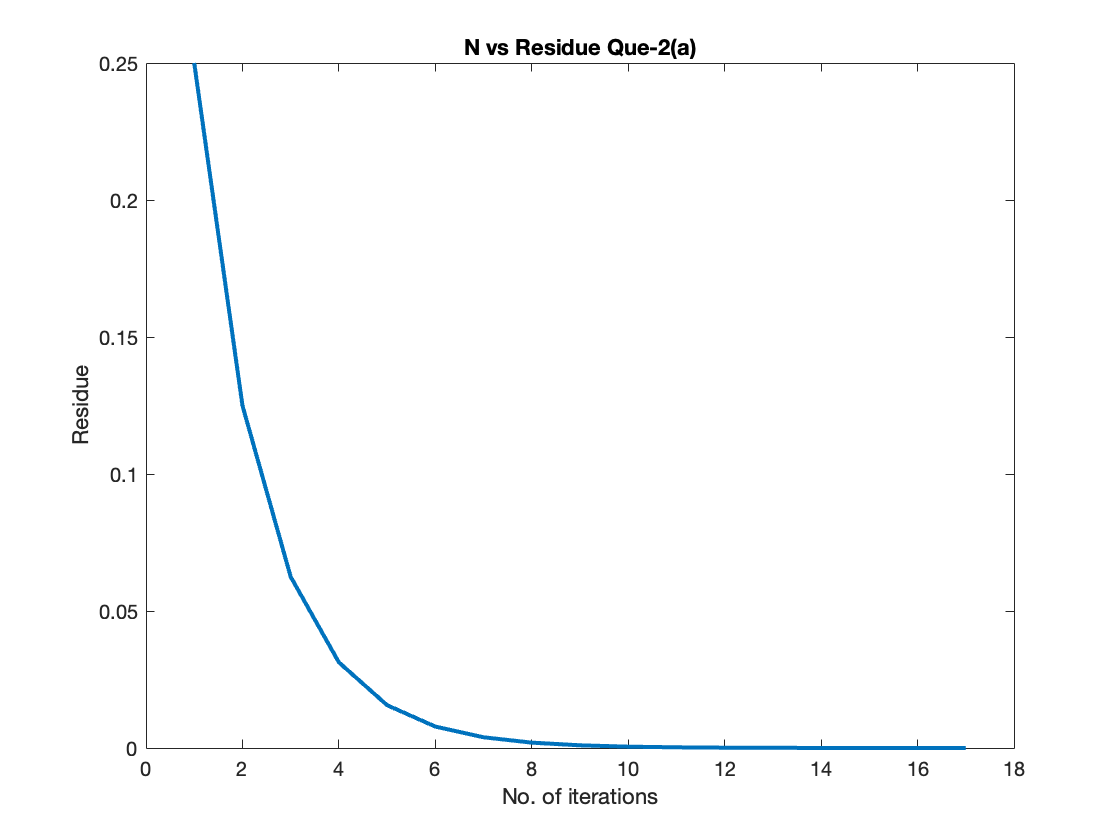
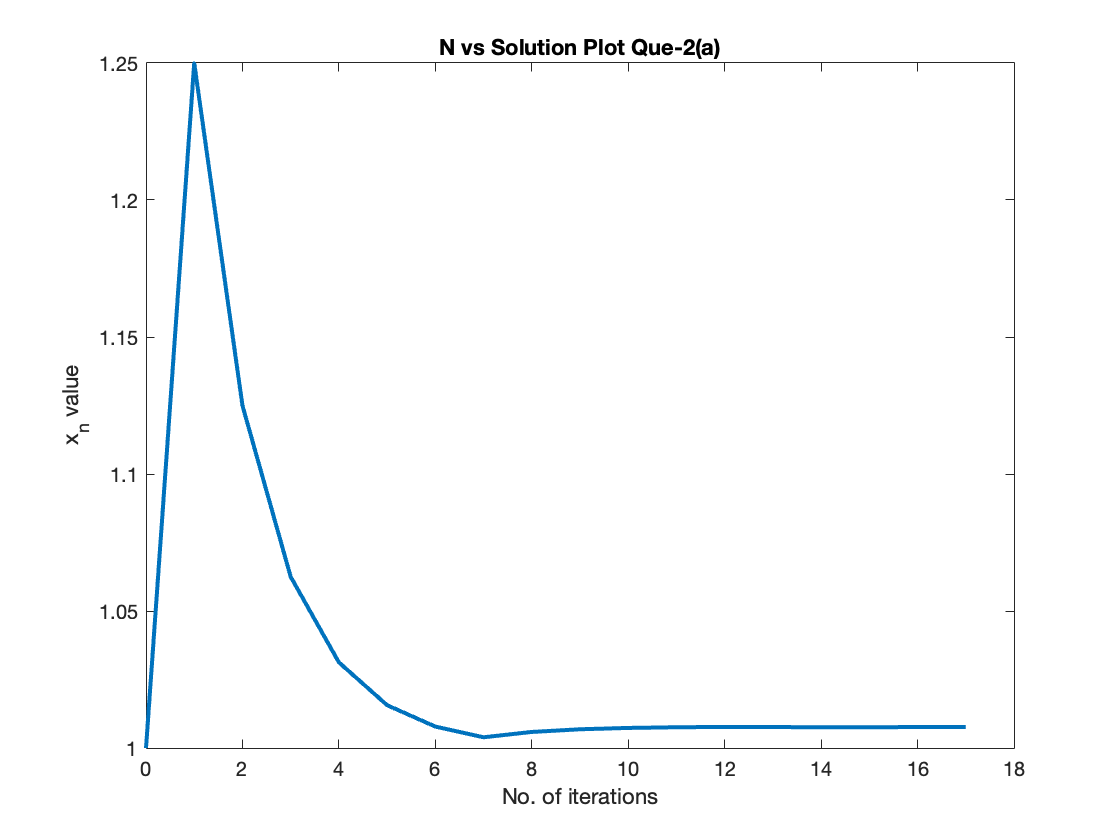
# Ques – 1

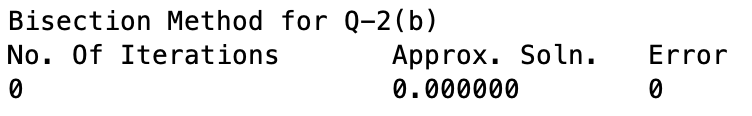


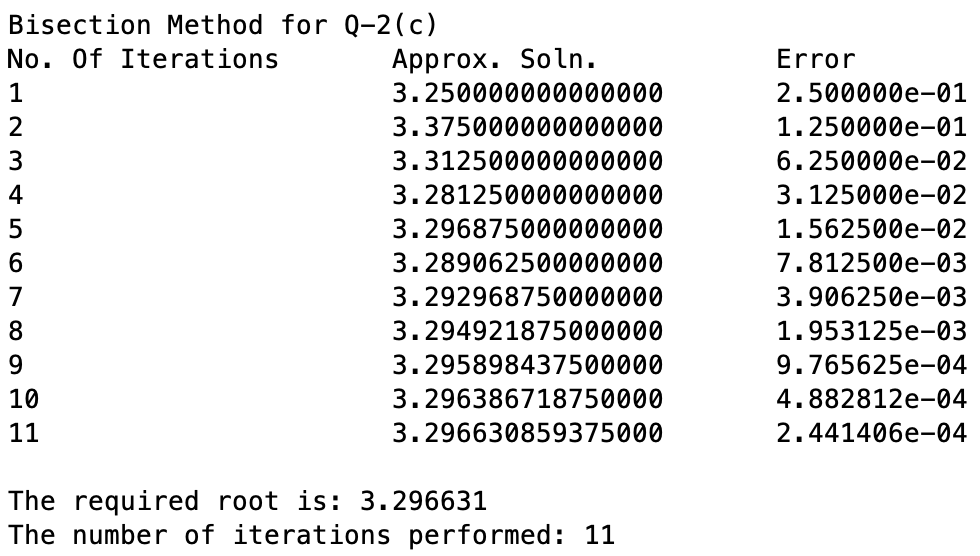


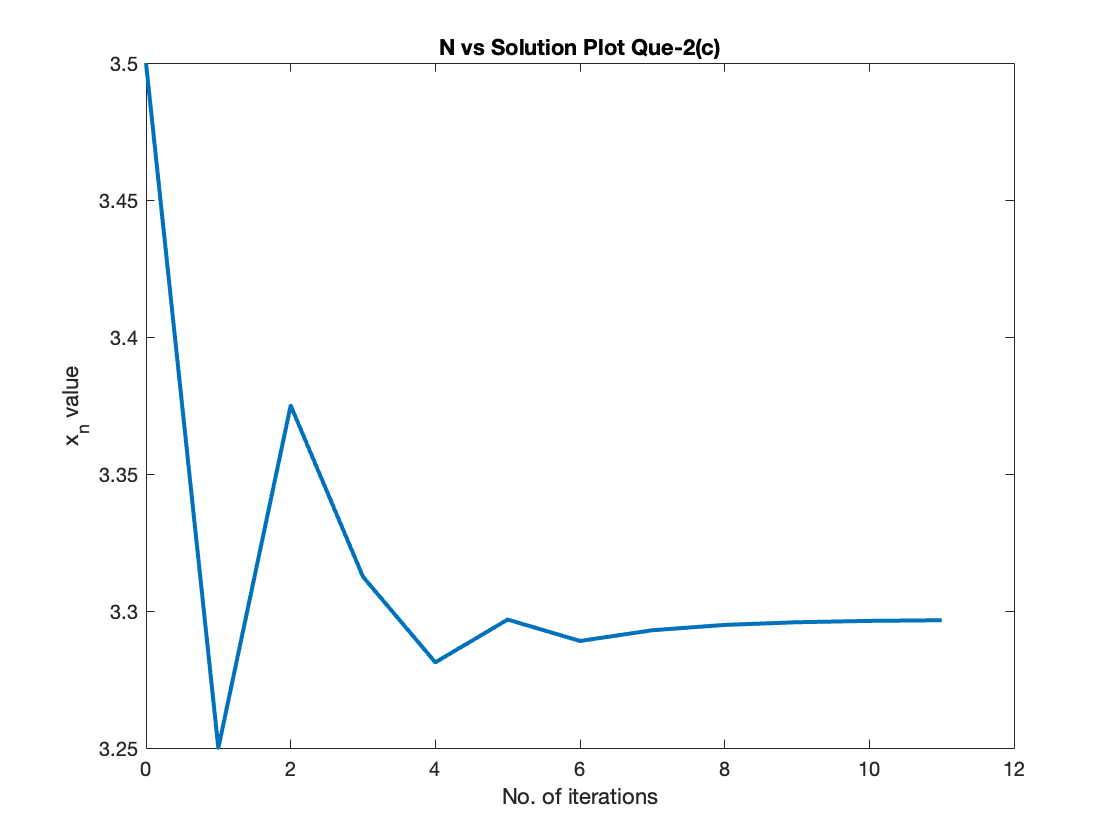
# Ques – 2

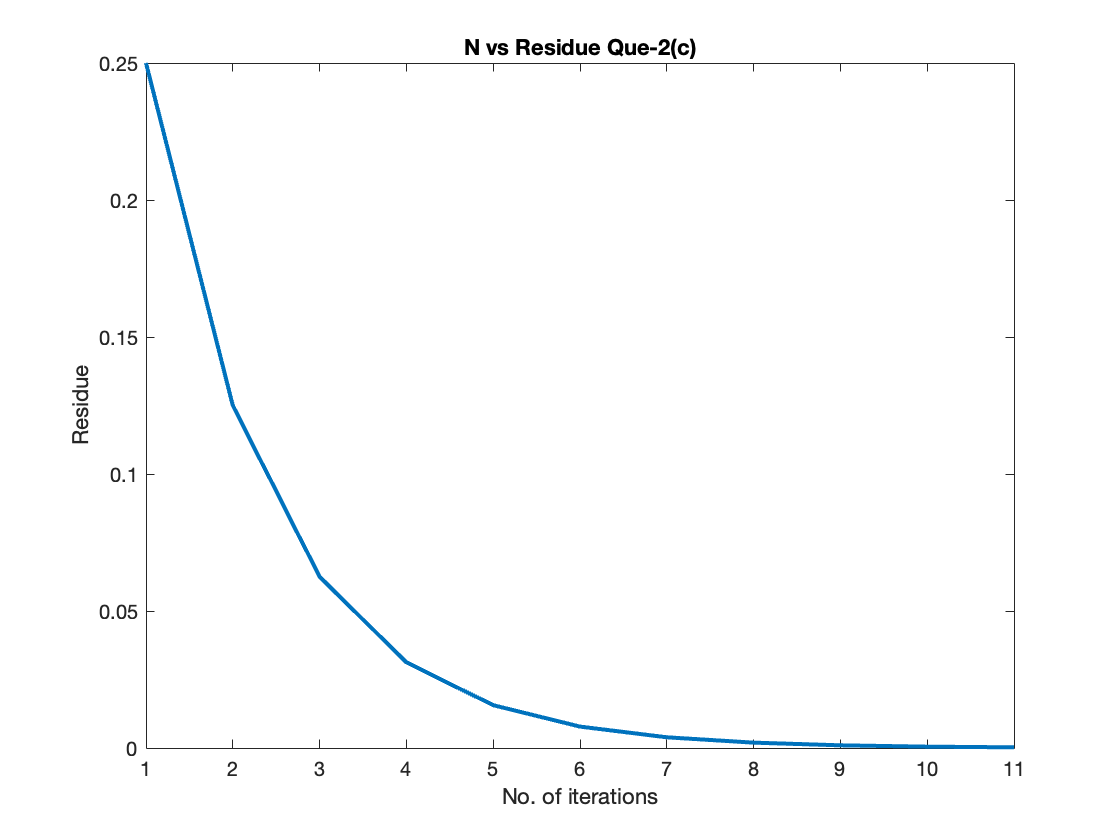




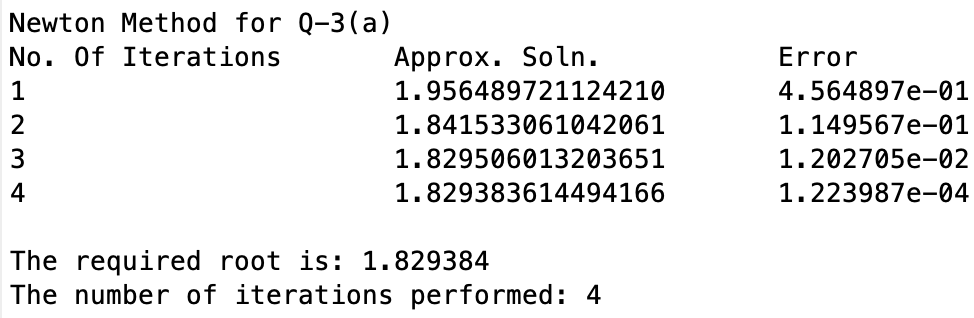


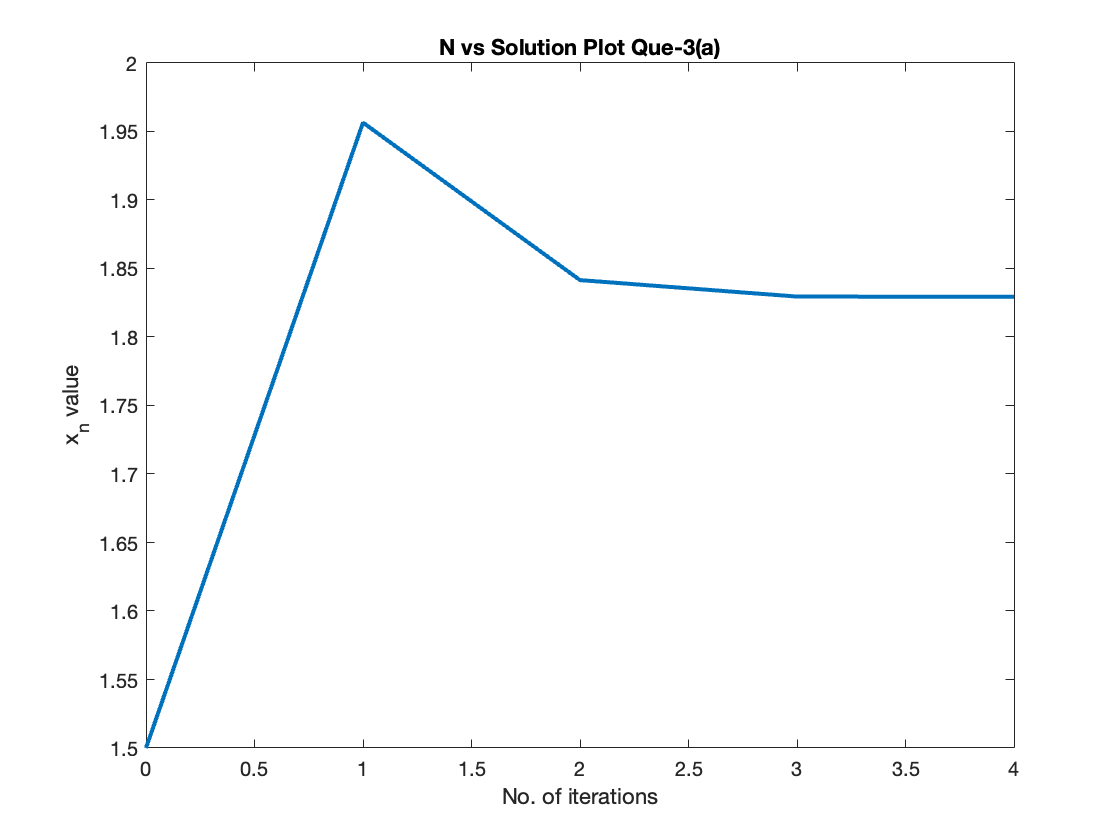


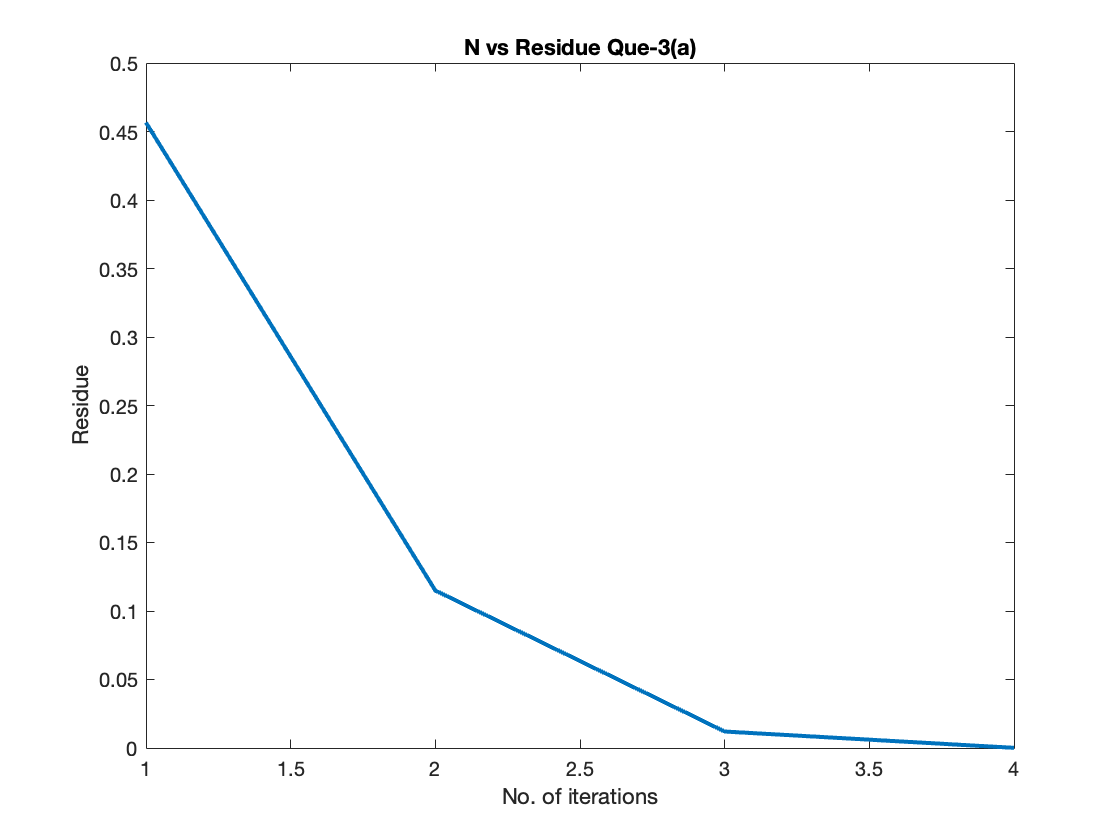


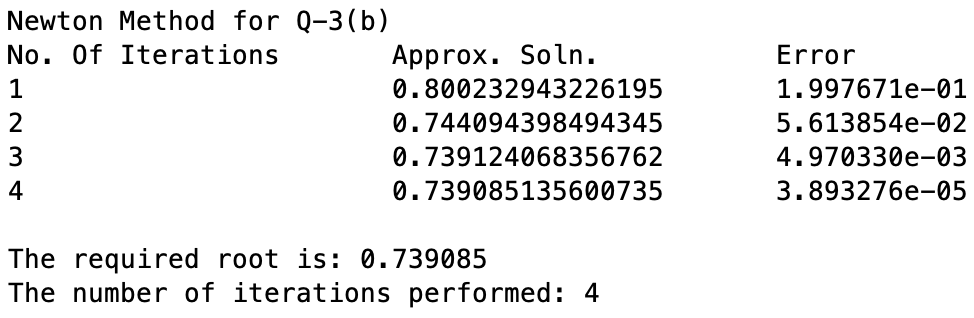


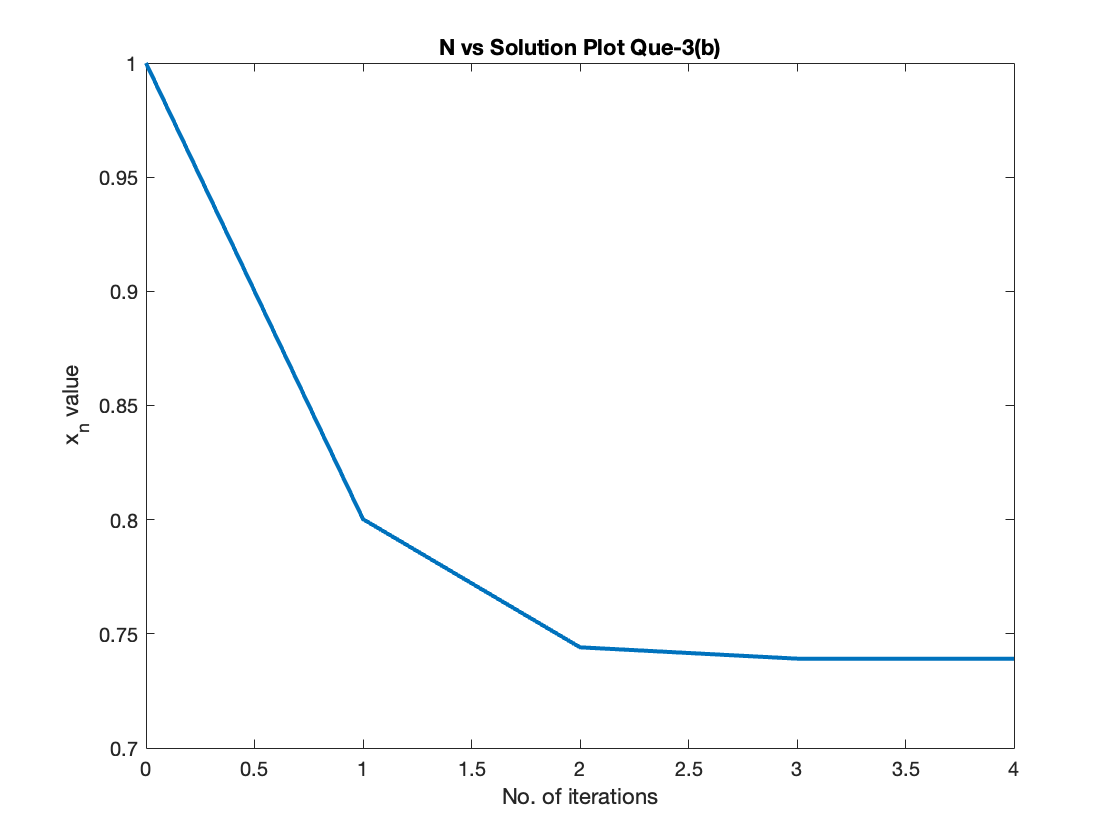
# Ques – 3

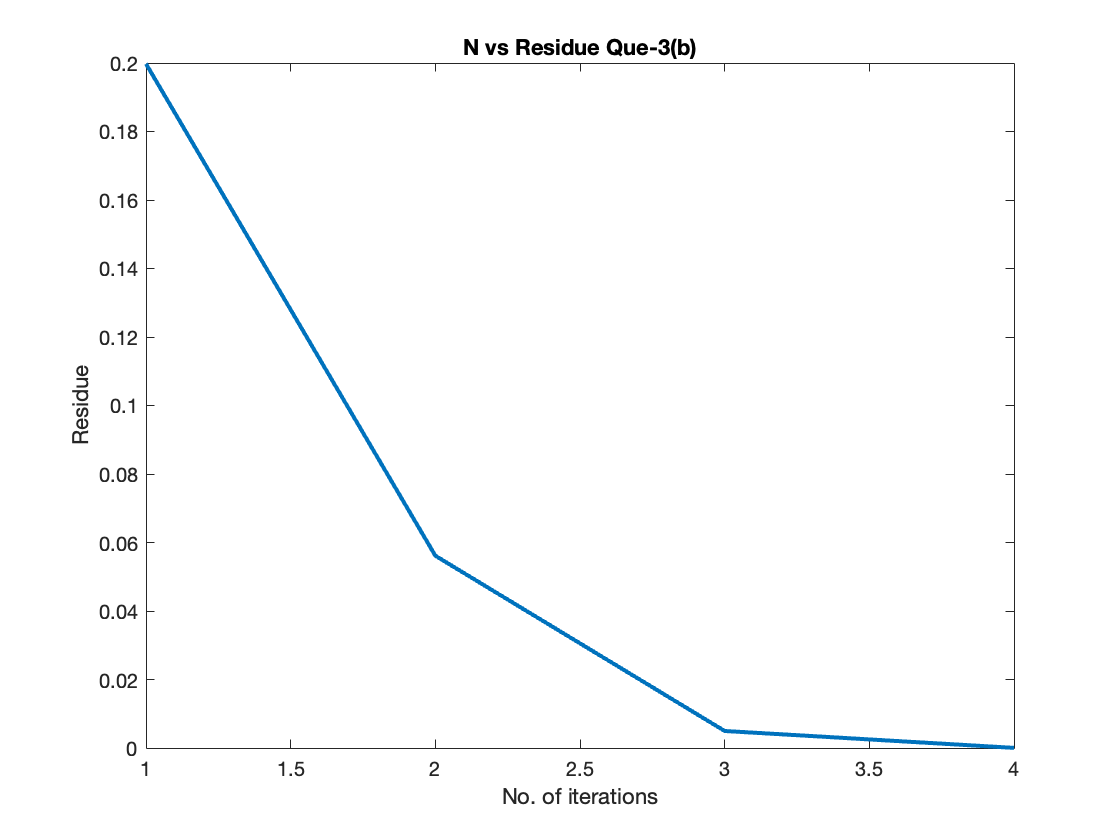


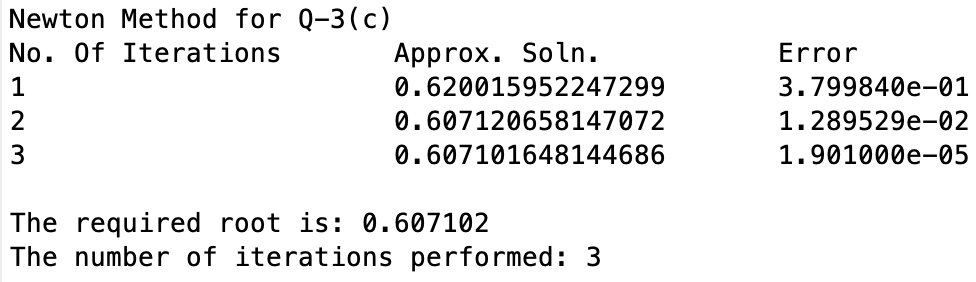


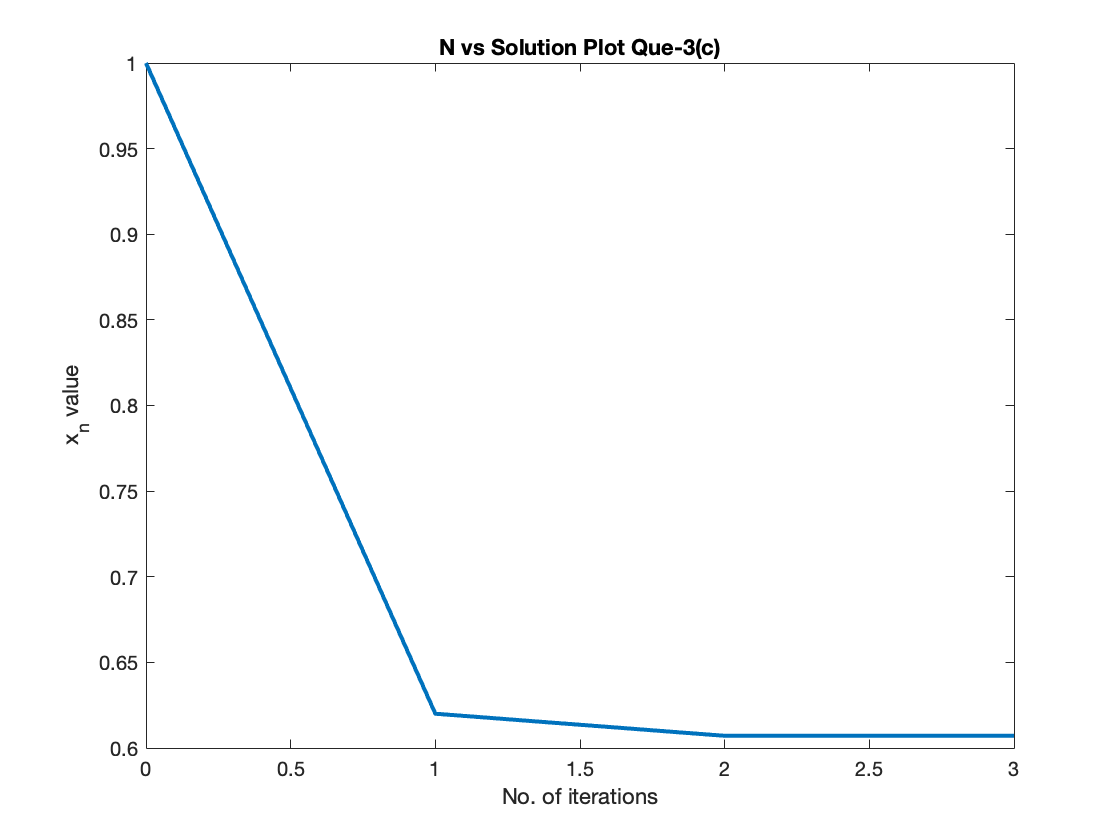
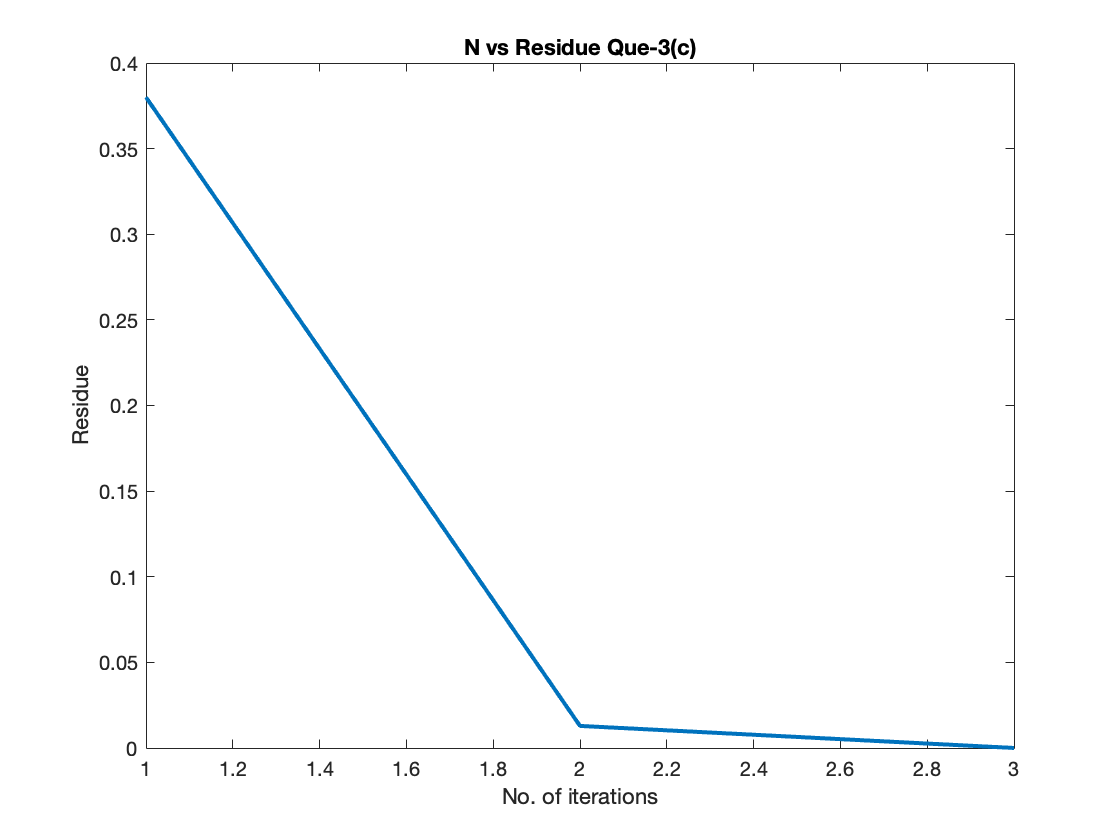


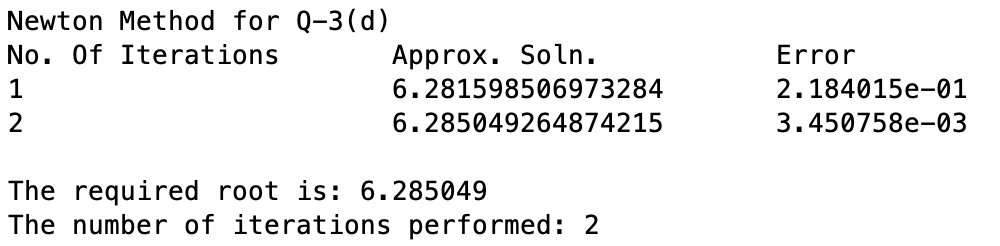


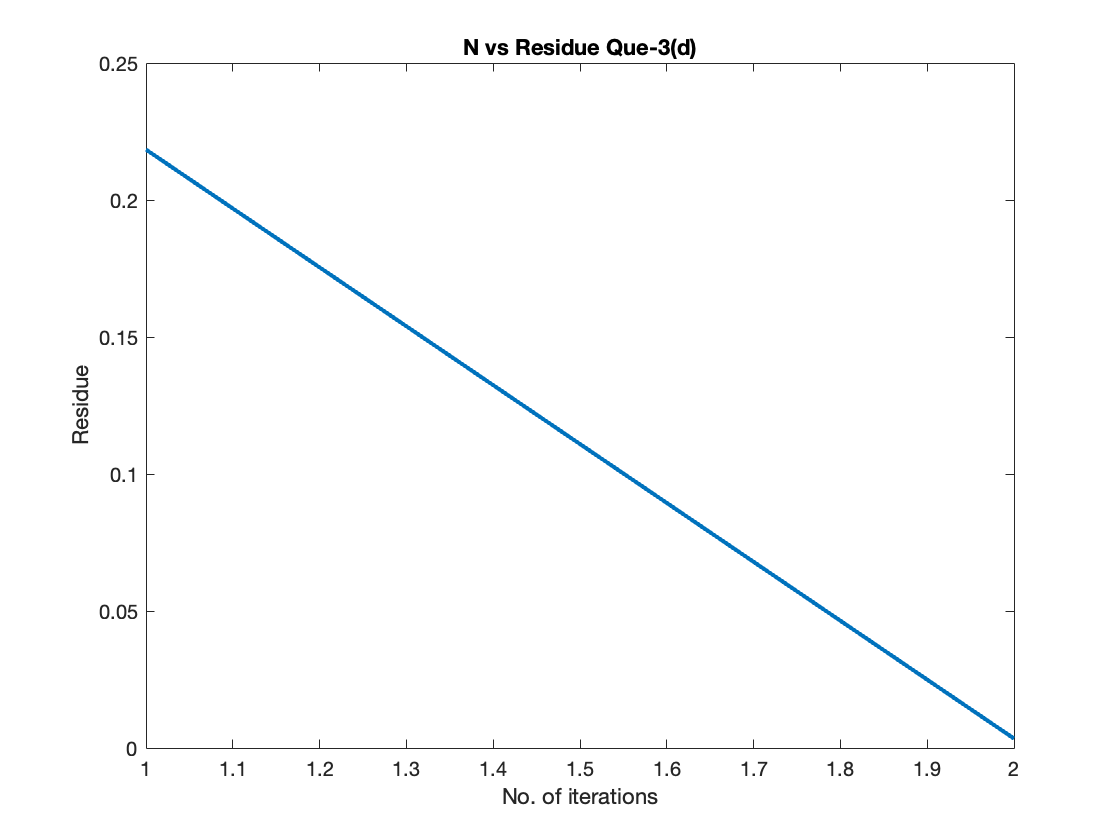
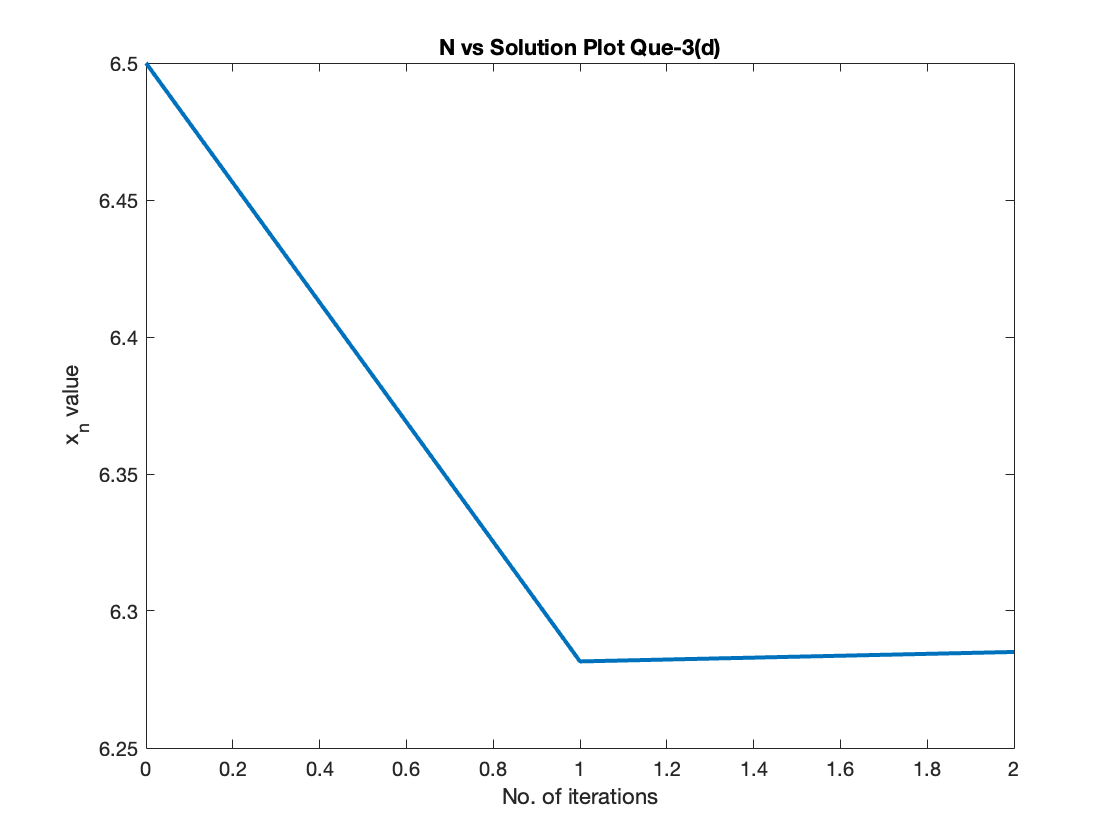






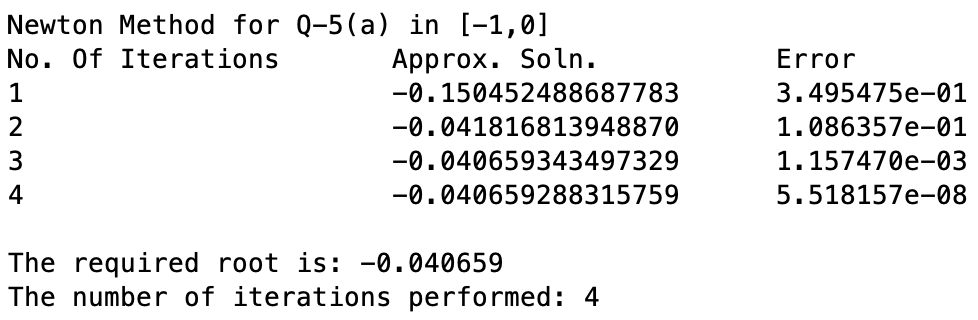


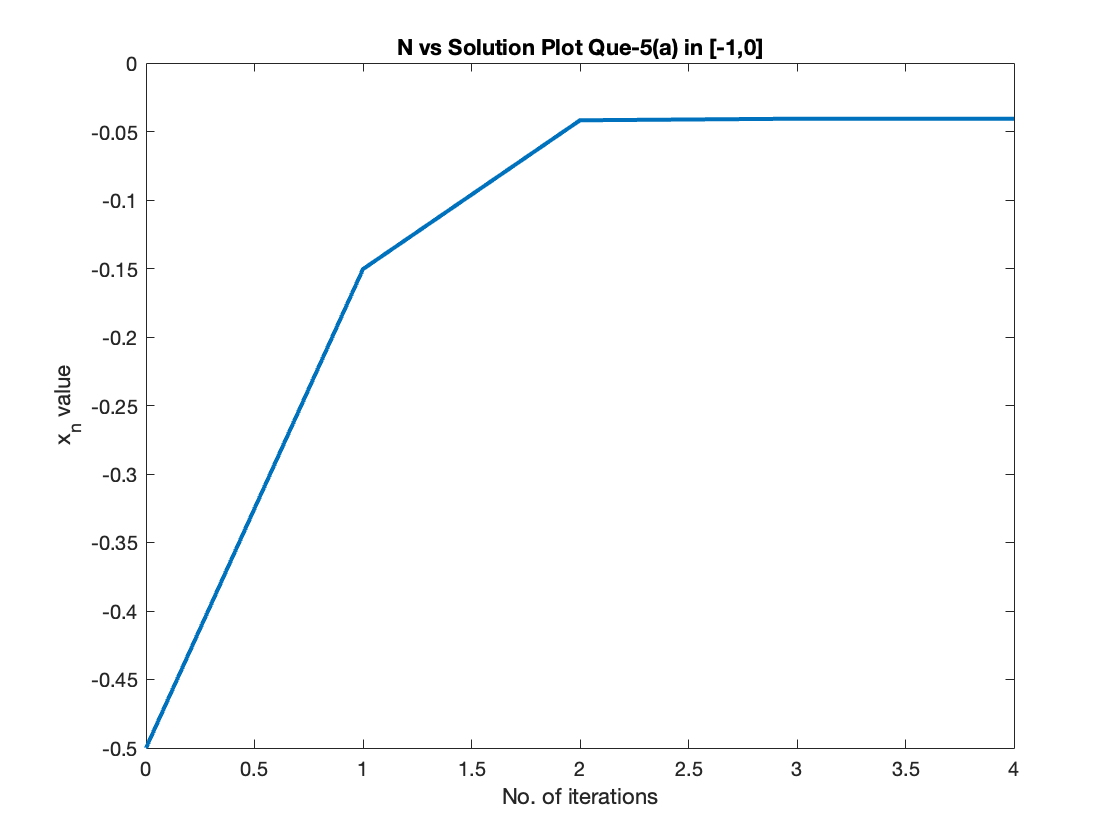


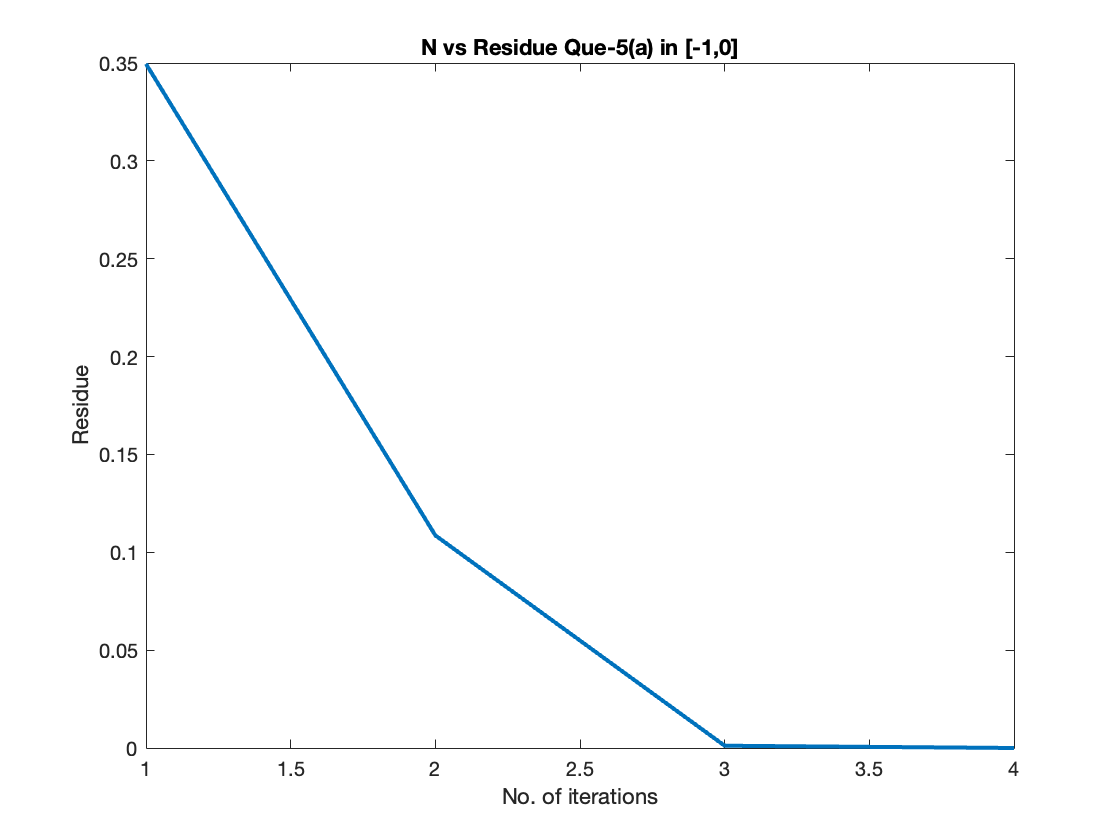


# Ques – 5

1. [-1,0]

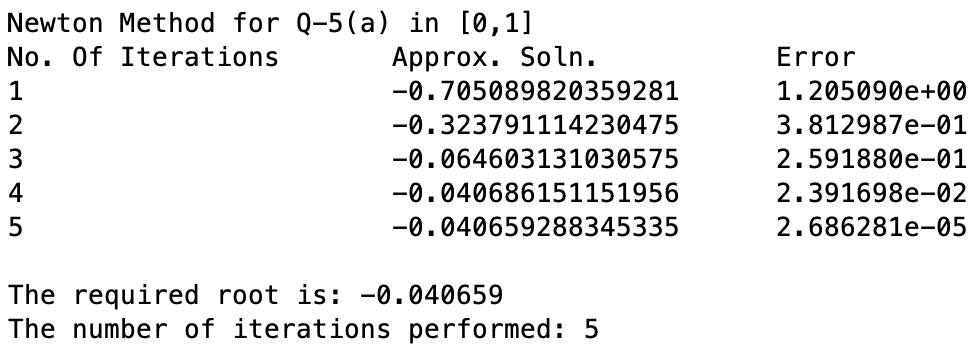


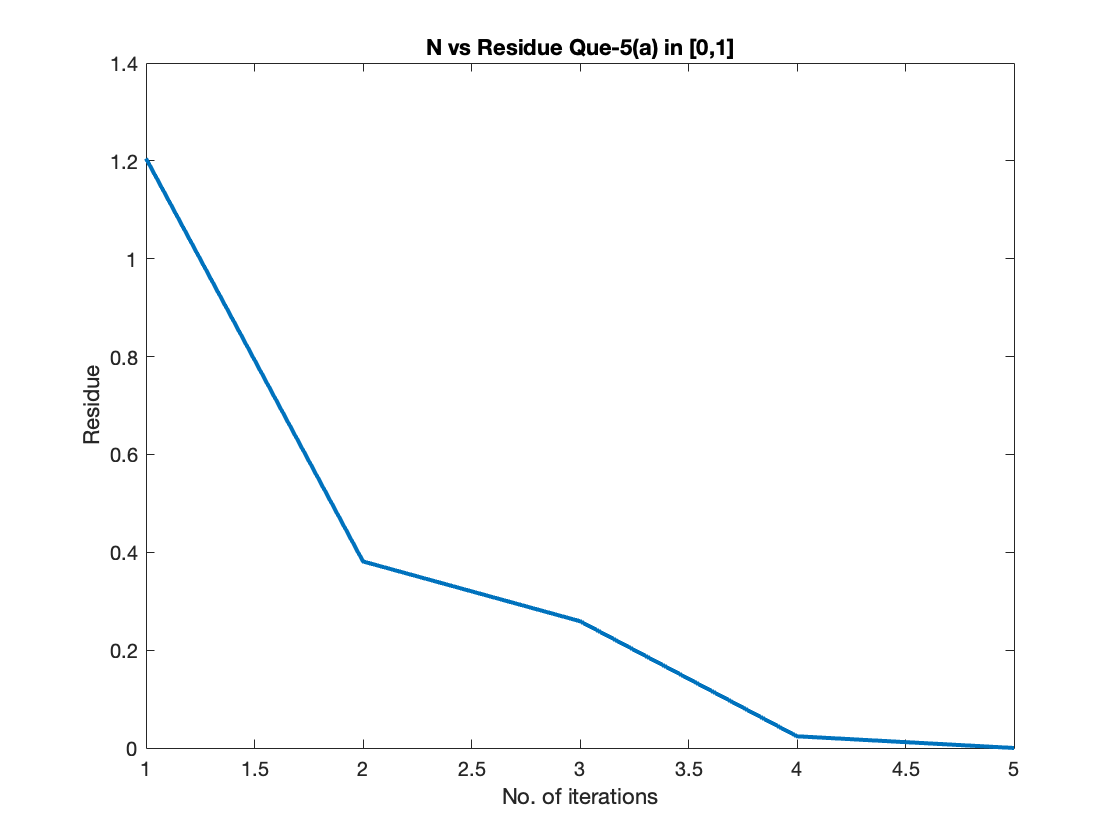
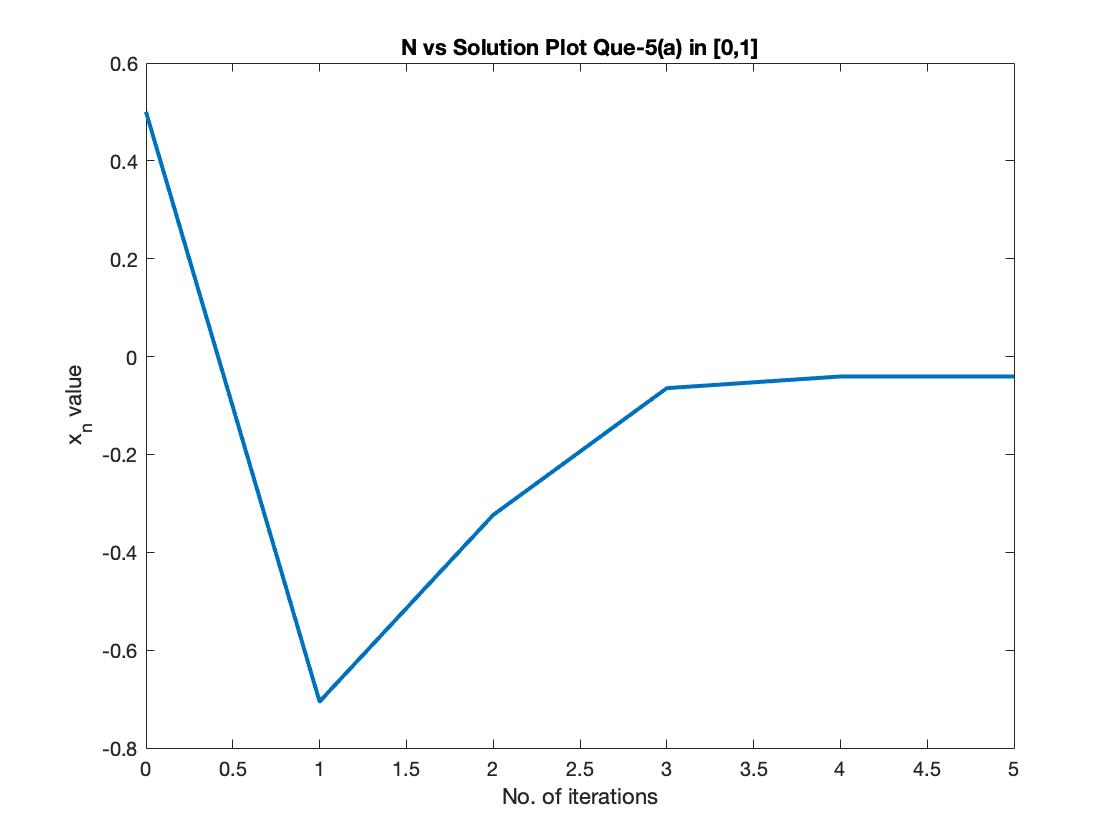




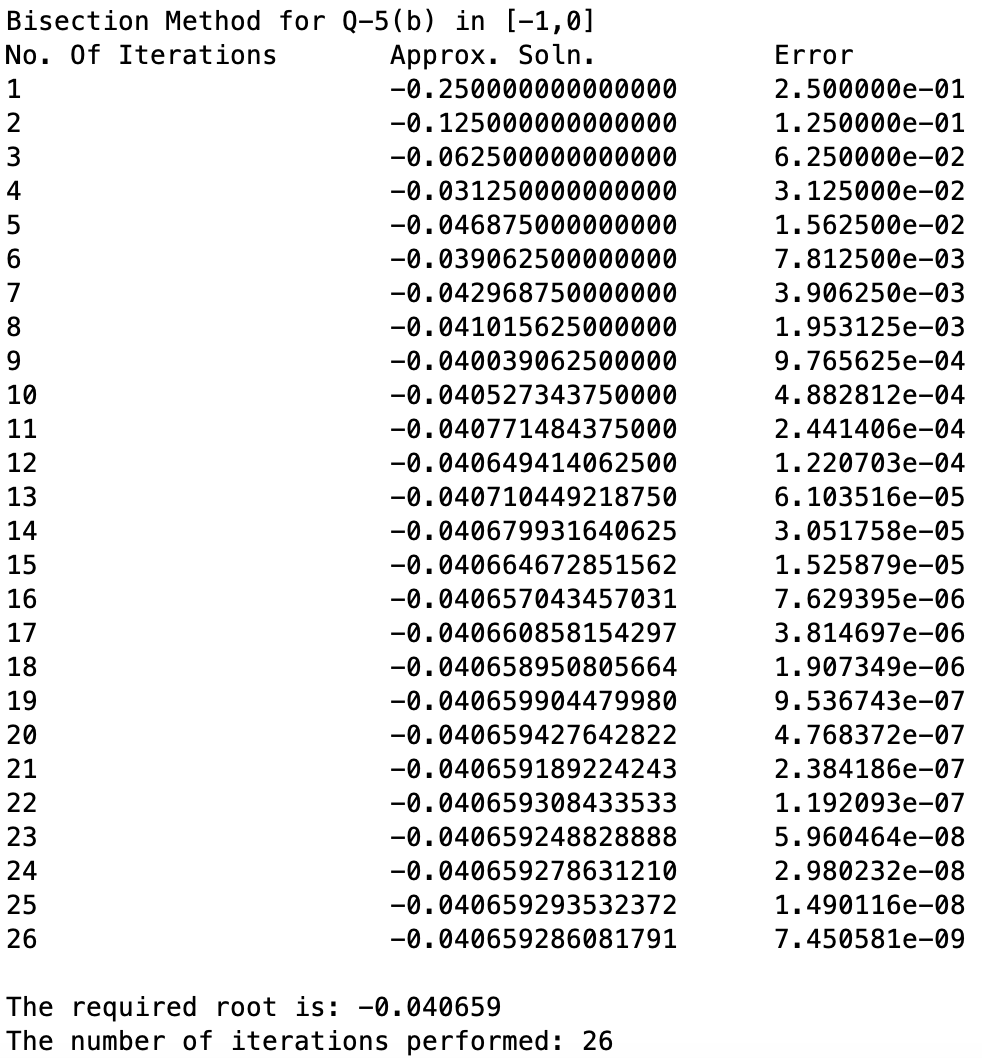
[0,1]

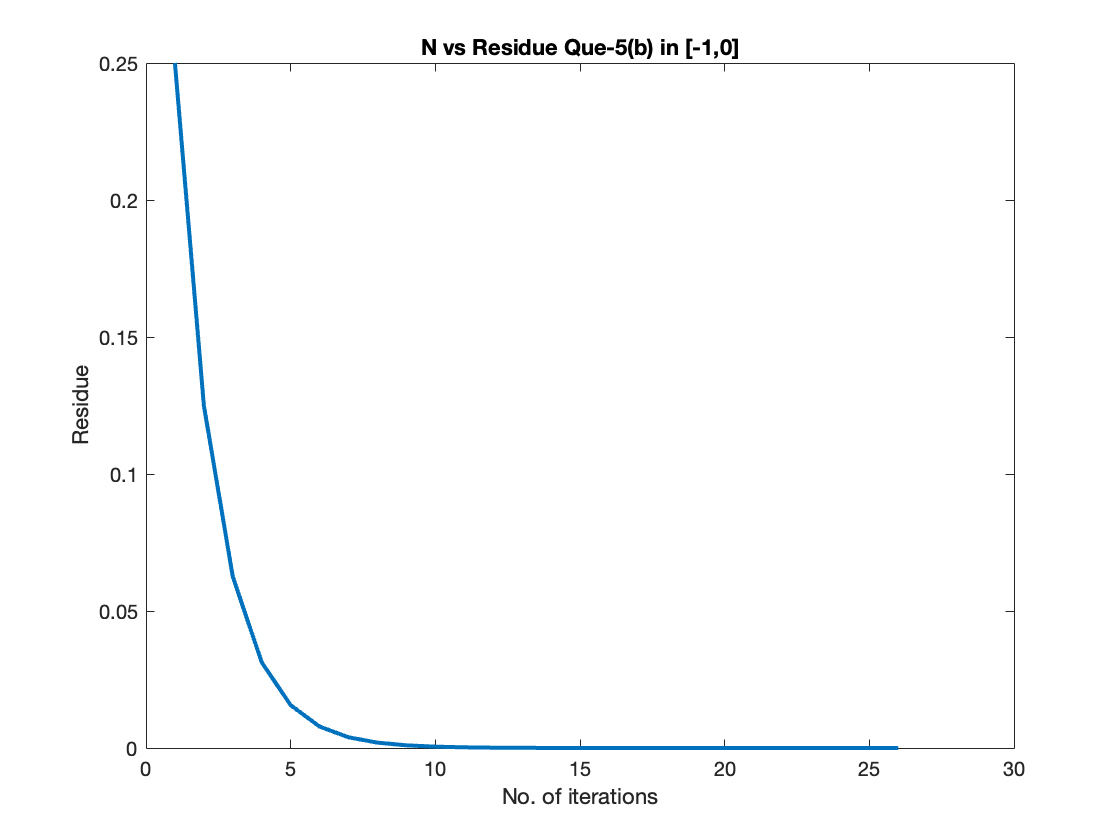
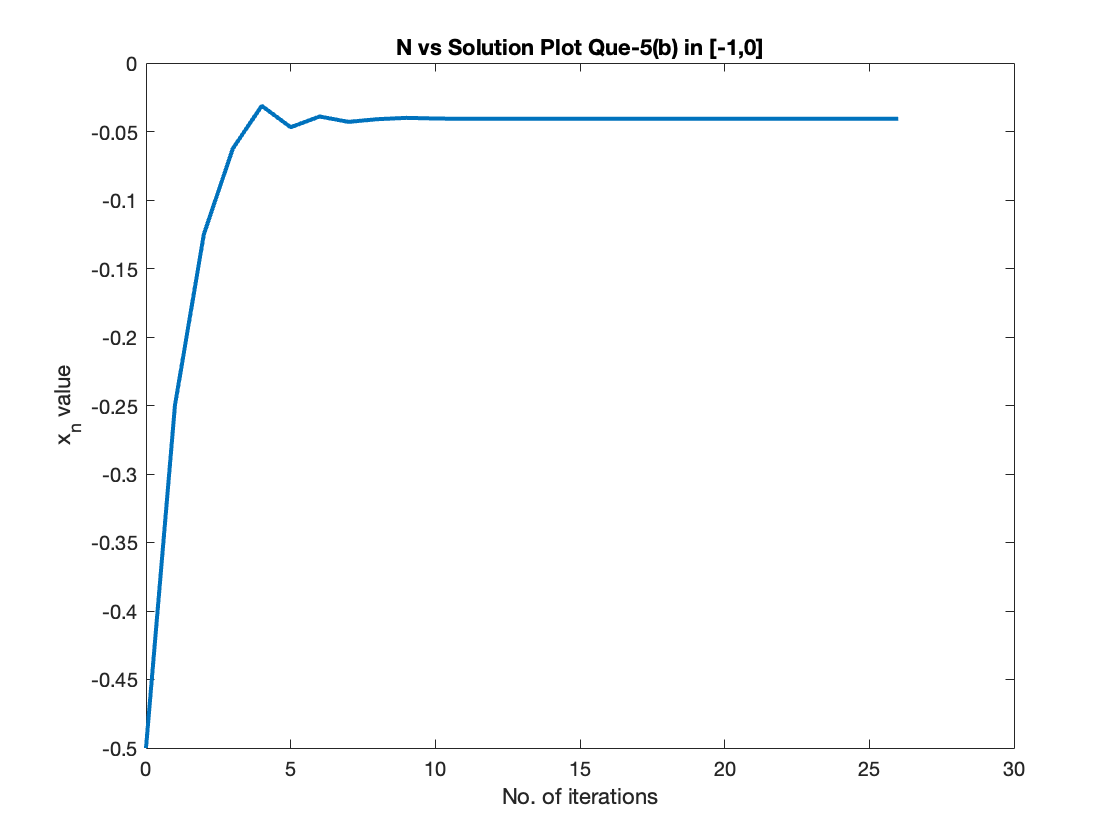
When calculating the root between [0,1] using Newton method by taking initial approximation as 0.5, the algorithm gives the root between [-1,0], this is because the derivative at 0.5 of the given function gives the next iterate as -0.705. Whereas using 0.6 as initial approximation, the algorithm will converge to the root between 0 and 1 which is approx. 0.962398



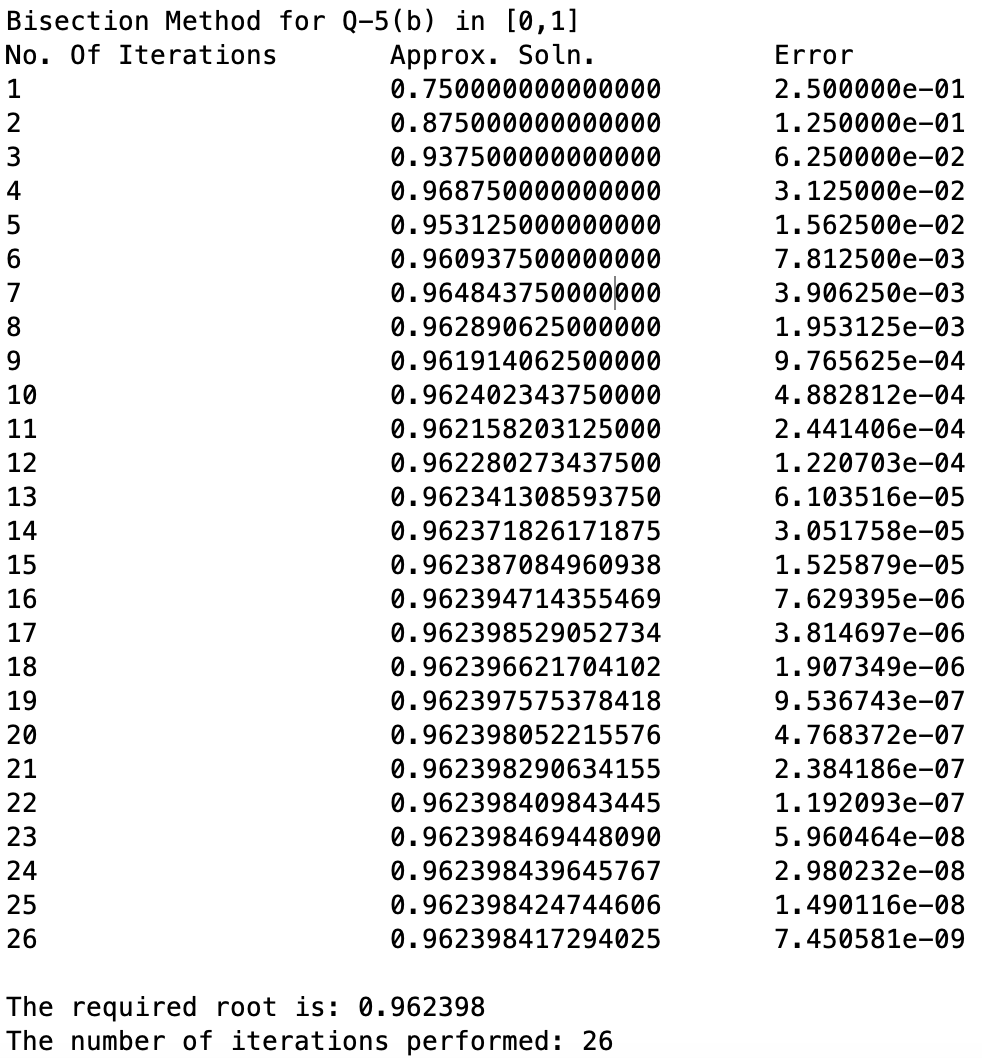


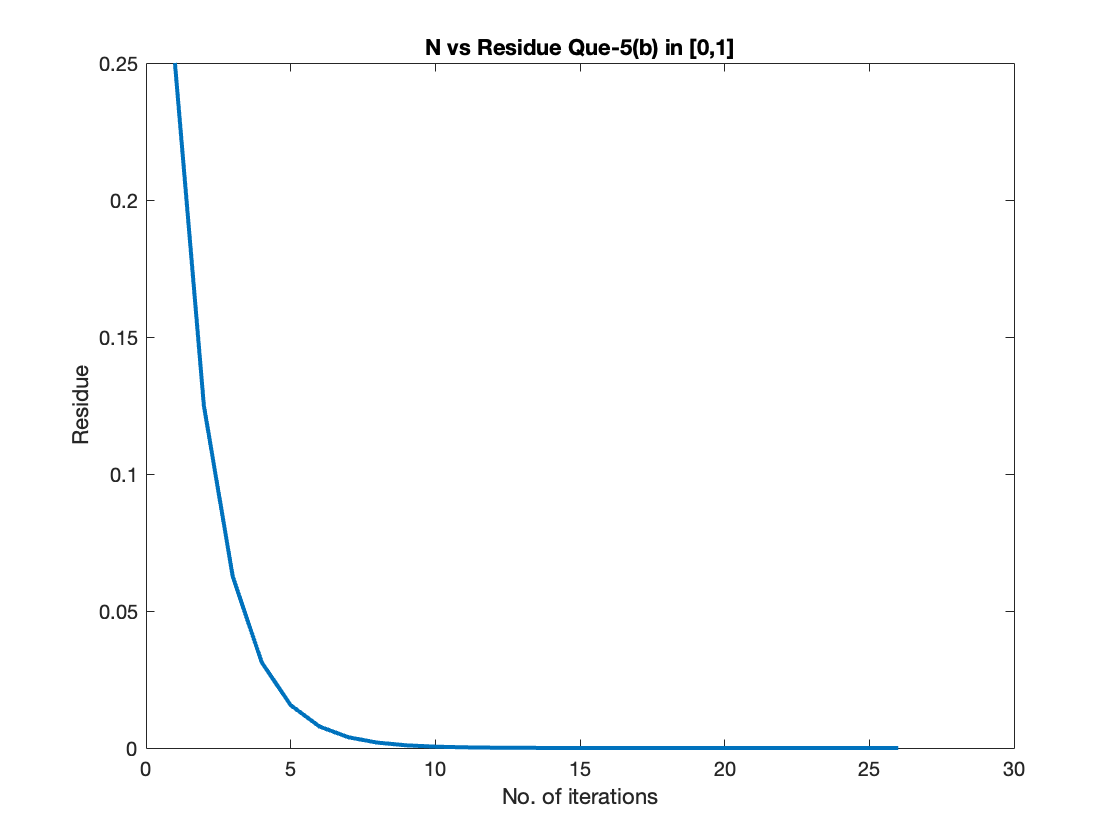
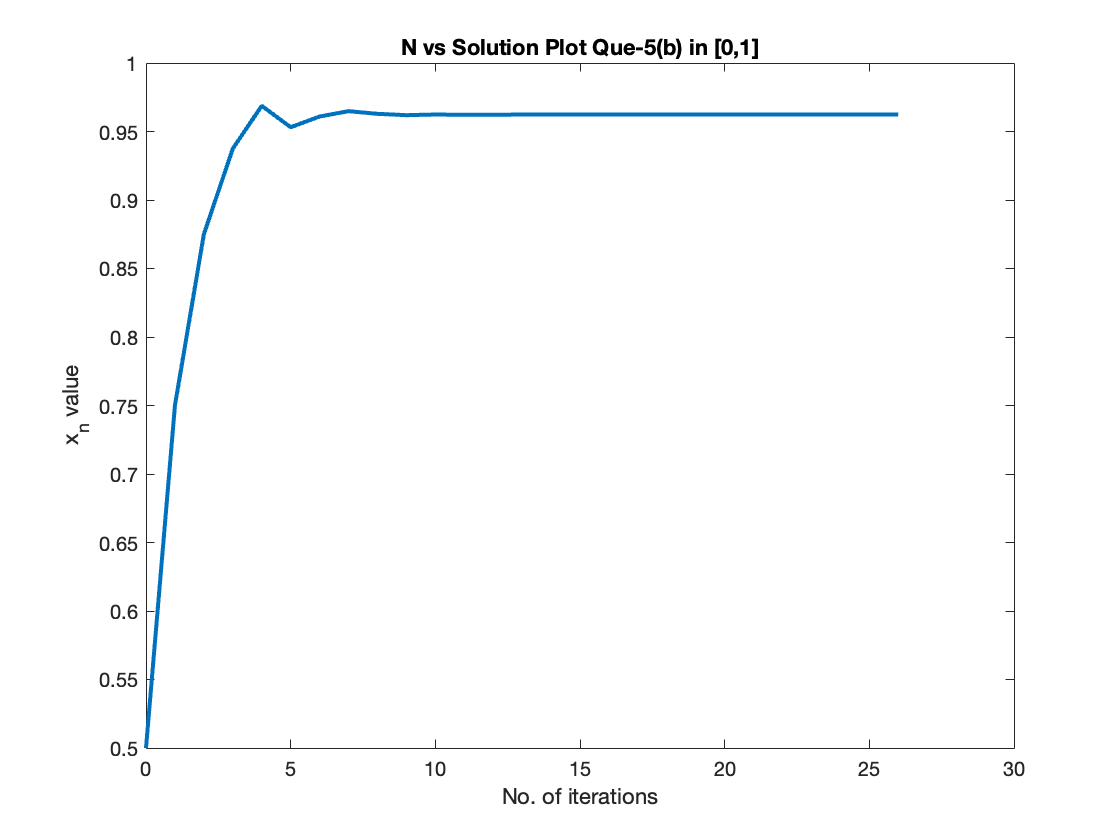
1. [-1,0]



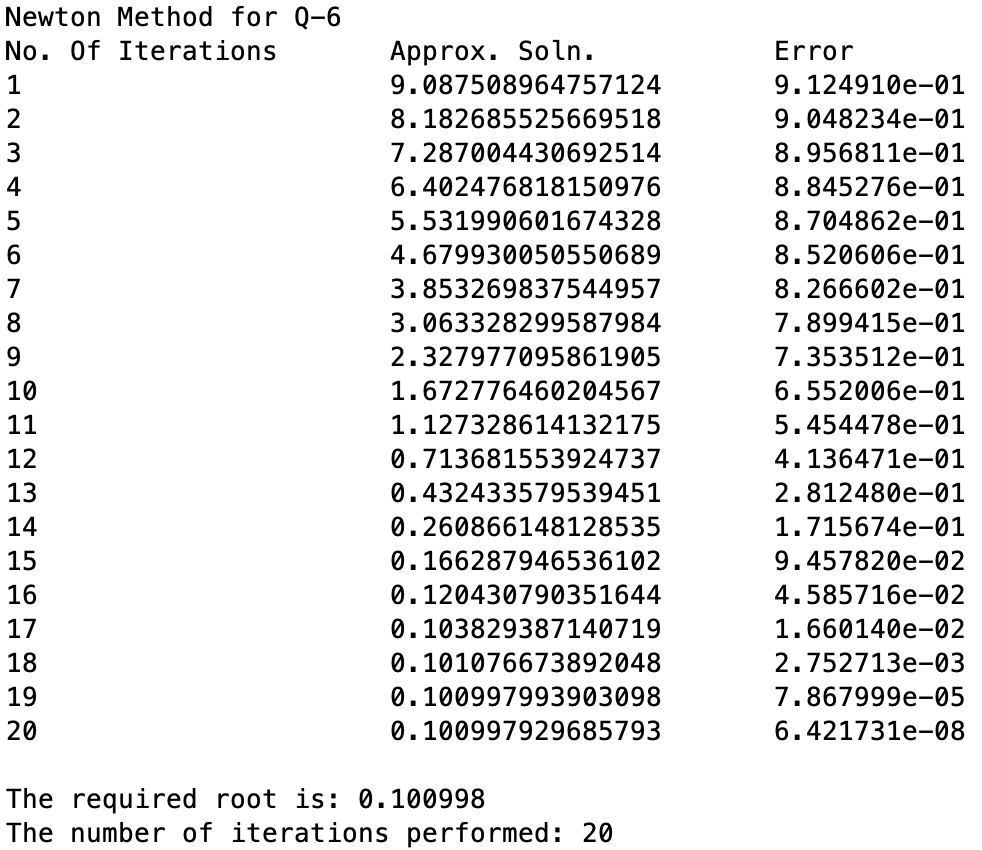


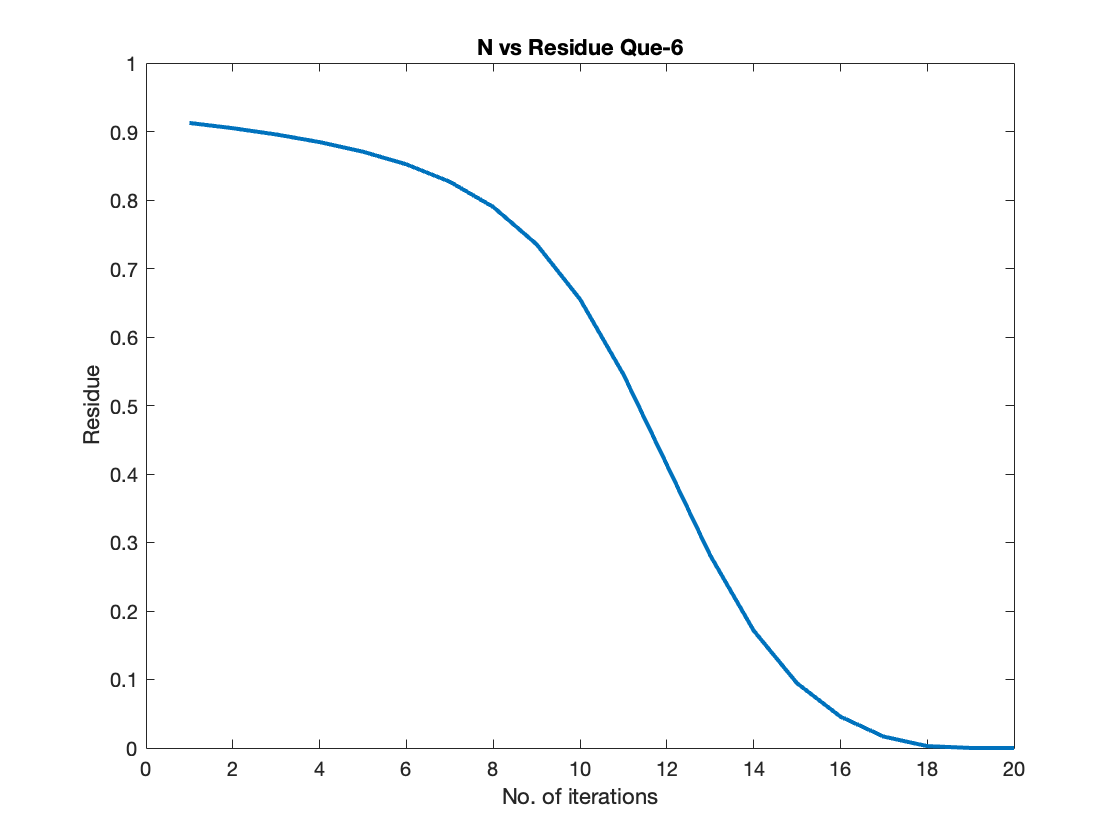
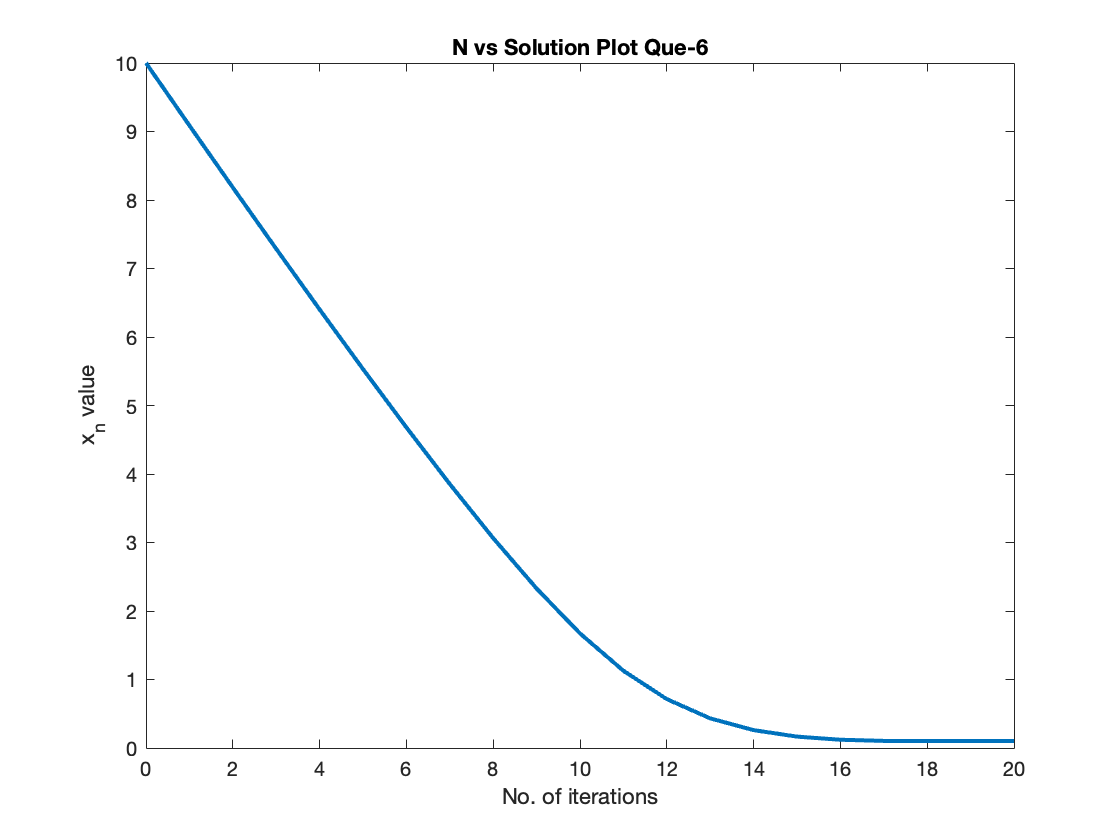
[0,1]



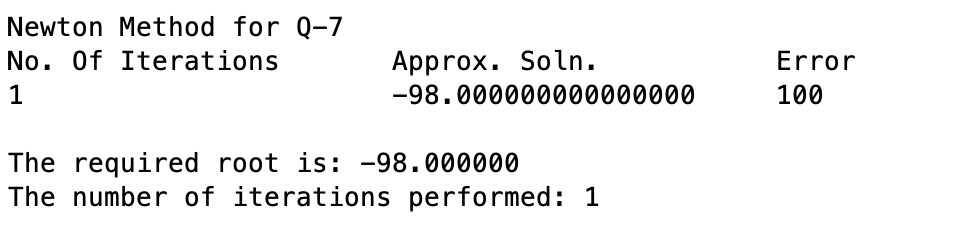


# Ques – 6



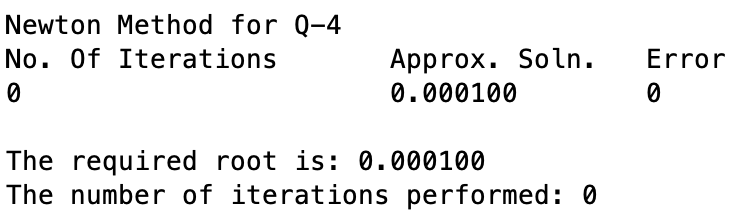


# Ques – 7



Newton’s method gives solution for the function at -98. This is due to the fact that the nature of the function changes at ~2.005 from decreasing to increasing due to which, the derivative at 2 gives the next iterate or newton’s method far from 2 which turns out to be -98 and the algorithm ends at it is one of the roots.

# Ques – 4



At x0 = 1e-4, the error is exp {-1 / (0.0001)2} = 6.451709693e-43429449 and

at x1 = 5e-5, the error is exp {-1/ (0.00005)2} = 1.732603825e-173717793.



The machine epsilon for the computer used is: 2.2204e-16

We can clearly observe that, at x0 = 0.0001 and x1 = 0.00005, the actual errors are much lesser than the machine epsilon, due to which, these values are accepted as zeros of the function though they are clearly not.

So, if we try to use newton method by taking initial approximation as 0.0001 or 0.00005, we won’t go any below of these, instead the algorithm will just declare the initial approximation as the root.

**Note: -** The machine epsilon is the smallest positive number that, when added to 1.0, results in a value different

from 1.0 in the floating-point representation of the computer.