MUHAMMAD UMAR KHAN
10619
NC LAB#05

```
def new_t(fn, x, tol=1e-8, n_iter=100):
    for i in range(n_iter):
        xnew=x- fn[0](x)/fn[1](x)
        if abs(xnew-x) < tol: break
        x=xnew
    return xnew, i

y=[lambda x: x*4 - x -10, lambda x: 4*x*3 - 1]
x, n=new_t(y, 1.5)
print("the root is %f at %d iteration" %(x,n))
```

```
the root is 3.333006 at 99 iteration
```

f=lambda x:x**3+4*(x**2)-10

f1=lambda x:x**2-3

f2=lambda x:x**3-x-1

f3=lambda x:x**3+1 import

math import sympy as eq

q=eq.symbols("x")

f=q**3+4*(q**2)-10

f1=q**2-3 f2=q**3-q-1

f3=q**3+1

der=eq.Derivative(f)

der=der.doit()

der1=eq.Derivative(f1)

der1=der1.doit()

der2=eq.Derivative(f2)

der2=der2.doit()

der3=eq.Derivative(f3)

der3=der3.doit()

MUHAMMAD UMAR KHAN
10619
NC LAB#05

```
print(der) print(der1)

print(der2)

print(der3)


fd=lambda x:3*x**2 + 8*x

fd1_=lambda x:2*x fd2=

lambda x:3*x**2 - 1 fd3=

lambda x:3*x**2 def

NewR(fun,fun1,x,to,niter):

for it in range(niter):


      a=x-(fun(x)/fun1(x))

if abs(a-x)<to:

break       else :

x=a


   return a,it x=float(input("enter x: ")) print("No#  Function        Derivative    interval

tol           no.  of  iterations    root")      tol=0.01  tol2=0.001  tol3=0.0001  niter=20

an,iterations=NewR(f,fd,x,tol,niter)   print("1    x**3+4*(x**2)-10  3*x**2 + 8*x   1.5

",tol,"         ",iterations,"            ",an)  an,iterations=NewR(f,fd,x,tol2,niter)   print("

x**3+4*(x**2)-10   3*x**2 + 8*x    1.5         ",tol2,"         ",iterations,"            ",an)

an,iterations=NewR(f,fd,x,tol3,niter)   print("    x**3+4*(x**2)-10  3*x**2 + 8*x   1.5

",tol3,"     ",iterations,"         ",an)


an,iterations=NewR(f1,fd1,x,tol,niter)

print("    x**2-3         2*x         1.5       ",tol,"       ",iterations,"         ",an)
```

MUHAMMAD UMAR KHAN
10619
NC LAB#05

```
an,iterations=NewR(f1,fd1,x,tol2,niter)

print("    x**2-3          2*x          1.5       ",tol2,"      ",iterations,"          ",an)

an,iterations=NewR(f1,fd1,x,tol3,niter)

print("    x**2-3          2*x          1.5       ",tol3,"     ",iterations,"          ",an)



an,iterations=NewR(f2,fd2,x,tol,niter)

print("3   x**3-x-1        3*x**2 - 1    1.5       ",tol,"       ",iterations,"          ",an)

an,iterations=NewR(f2,fd2,x,tol2,niter)

print("    x**3-x-1        3*x**2 - 1    1.5       ",tol2,"      ",iterations,"          ",an)

an,iterations=NewR(f2,fd2,x,tol3,niter)

print("    x**3-x-1        3*x**2 - 1    1.5       ",tol3,"     ",iterations,"          ",an)
an,iterations=NewR(f3,fd3,x,tol,niter)

print("4   x**3+1          3*x**2        1.5       ",tol,"       ",iterations,"          ",an)

an,iterations=NewR(f3,fd3,x,tol2,niter)

print("    x**3+1          3*x**2        1.5       ",tol2,"      ",iterations,"          ",an)

an,iterations=NewR(f3,fd3,x,tol3,niter)

print("    x**3+1          3*x**2        1.5       ",tol3,"     ",iterations,"          ",an)
```

MUHAMMAD UMAR KHAN
10619
NC LAB#05

jupyter task 5 Last Checkpoint: a minute ago (autosaved)

File Edit View Insert Cell Kernel Widgets Help Not Trusted

Run Code

In [4]:
```
f=lambda x:x**3+4*(x**2)-10
f1=lambda x:x**2-3
f2=lambda x:x**3-x-1
f3=lambda x:x**3+1
import math
import sympy as eq
q=eq.symbols("x")
f=q**3+4*(q**2)-10
f1=q**2-3
f2=q**3-q-1
f3=q**3+1

der=eq.Derivative(f)
der=der.doit()
der1=eq.Derivative(f1)
der1=der1.doit()
der2=eq.Derivative(f2)
der2=der2.doit()
der3=eq.Derivative(f3)
der3=der3.doit()

print(der)
print(der1)
print(der2)
print(der3)

fd=lambda x:3*x**2 + 8*x
fd1_=lambda x:2*x
fd2= lambda x:3*x**2 - 1
fd3= lambda x:3*x**2
```

```
3*x**2 + 8*x
2*x
3*x**2 - 1
3*x**2
```

In [5]:
```
def NewR(fun,fun1,x,to,niter):
    for it in range(niter):

        a=x-(fun(x)/fun1(x))
        if abs(a-x)<to:
            break
        else :
            x=a

    return a,it
```

In [6]:
```
x=float(input("enter x: "))
print("No#  Function         Derivative     interval    tol       no. of iterations  root")
tol=0.01
tol2=0.001
tol3=0.0001
niter=20
an,iterations=NewR(f,fd,x,tol,niter)
print("1    x**3+4*(x**2)-10  3*x**2 + 8*x   1.5        ",tol,"       ",iterations,"          ",an)
an,iterations=NewR(f,fd,x,tol2,niter)
print("     x**3+4*(x**2)-10  3*x**2 + 8*x   1.5        ",tol2,"      ",iterations,"          ",an)
an,iterations=NewR(f,fd,x,tol3,niter)
print("     x**3+4*(x**2)-10  3*x**2 + 8*x   1.5        ",tol3,"     ",iterations,"          ",an)

an,iterations=NewR(f1,fd1,x,tol,niter)
print("     x**2-3            2*x            1.5        ",tol,"       ",iterations,"          ",an)
an,iterations=NewR(f1,fd1,x,tol2,niter)
print("     x**2-3            2*x            1.5        ",tol2,"      ",iterations,"          ",an)
an,iterations=NewR(f1,fd1,x,tol3,niter)
print("     x**2-3            2*x            1.5        ",tol3,"     ",iterations,"          ",an)

an,iterations=NewR(f2,fd2,x,tol,niter)
print("3    x**3-x-1          3*x**2 - 1     1.5        ",tol,"       ",iterations,"          ",an)
an,iterations=NewR(f2,fd2,x,tol2,niter)
print("     x**3-x-1          3*x**2 - 1     1.5        ",tol2,"      ",iterations,"          ",an)
an,iterations=NewR(f2,fd2,x,tol3,niter)
print("     x**3-x-1          3*x**2 - 1     1.5        ",tol3,"     ",iterations,"          ",an)

an,iterations=NewR(f3,fd3,x,tol,niter)
print("4    x**3+1            3*x**2         1.5        ",tol,"       ",iterations,"          ",an)
an,iterations=NewR(f3,fd3,x,tol2,niter)
print("     x**3+1            3*x**2         1.5        ",tol2,"      ",iterations,"          ",an)
an,iterations=NewR(f3,fd3,x,tol3,niter)
print("     x**3+1            3*x**2         1.5        ",tol3,"     ",iterations,"          ",an)
```

MUHAMMAD UMAR KHAN
10619
NC LAB#05

```
enter x: 1.5
No#  Function          Derivative     interval    tol       no. of iterations  root
1    x**3+4*(x**2)-10  3*x**2 + 8*x   1.5         0.01          1              1.3652620148746266
     x**3+4*(x**2)-10  3*x**2 + 8*x   1.5         0.001         2              1.3652300139161466
     x**3+4*(x**2)-10  3*x**2 + 8*x   1.5         0.0001        2              1.3652300139161466
     x**2-3            2*x            1.5         0.01          2              1.7320508100147276
     x**2-3            2*x            1.5         0.001         2              1.7320508100147276
     x**2-3            2*x            1.5         0.0001        2              1.7320508100147276
3    x**3-x-1          3*x**2 - 1     1.5         0.01          2              1.3247181739990537
     x**3-x-1          3*x**2 - 1     1.5         0.001         2              1.3247181739990537
     x**3-x-1          3*x**2 - 1     1.5         0.0001        3              1.3247179572447898
4    x**3+1            3*x**2         1.5         0.01          13              -1.000005923086305
     x**3+1            3*x**2         1.5         0.001         14              -1.0000000000350826
     x**3+1            3*x**2         1.5         0.0001        14              -1.0000000000350826
```

S No.	Functions	Starting Interval	Tolerance	Total Iterations	Root
1.	x^3–x–1=0	1.5	0.01	2	1.324718
			0.001	2	1.324718
			0.0001	3	1.324718
2.	x^3+1=0	1.5	0.01	13	-1.000005
			0.001	14	-1.000000
			0.0001	14	-1.000000
3.	x^2-3=0	1.5	0.01	2	1.732051
			0.001	2	1.732051
			0.0001	2	1.732051
4.	x^3+4*x^2-10=0	1.5	0.01	1	1.365262
			0.001	2	1.365230
			0.0001	2	1.365230