ASSIGNMENT-5

Q1.

i.)

```
>>> from math import *
>>> from sympy import *
>>> from numpy import *
>>> def simpsons13(f,a,b,n):
...     h=float(b-a)/n
...     result = f(a) + f(b)
...     for i in range(1,n):
...         k= a + i*h
...         if i%2==0:
...             result=result+2*f(k)
...         else:
...             result=result+4*f(k)
...     result *=h/3
...     return result
...
>>> def f(x):
...     return sin(x)
...
>>> simpsons13(f,0,pi,6)
2.0008631896735363
```

ii.)

```
>>> from math import *
>>> from numpy import *
>>> def simpsons13(f,a,b,n):
...     h=float(b-a)/n
...     result = f(a) + f(b)
...     for i in range(1,n):
...         k= a + i*h
...         if i%2==0:
```

```
...         result=result+2*f(k)
...       else:
...         result=result+4*f(k)
...   result *=h/3
...   return result
...
>>> def f(x):
...   return 1/1+x**2
...
>>> simpsons13(f,0,1,4)
1.3333333333333333
```

iii.)

```
>>> from math import *
>>> from numpy import *
>>> def simpsons13(f,a,b,n):
...   h=float(b-a)/n
...   result = f(a) + f(b)
...   for i in range(1,n):
...     k= a + i*h
...     if i%2==0:
...       result=result+2*f(k)
...     else:
...       result=result+4*f(k)
...   result *=h/3
...   return result
...
>>> def f(x):
...   return x*sin(x)
...
>>> simpsons13(f,0,pi,6)
2.8409945748603387
```

iv.)

```
>>> from math import *
```

```
>>> from numpy import *
>>> def simpsons13(f,a,b,n):
...     h=float(b-a)/n
...     result = f(a) + f(b)
...     for i in range(1,n):
...         k= a + i*h
...         if i%2==0:
...             result=result+2*f(k)
...         else:
...             result=result+4*f(k)
...     result *=h/3
...     return result
...
>>> def f(x):
...     return x**3
...
>>> simpsons13(f,1,5,6)
155.99999999999997
```

v.)

```
>>> from math import *
>>> from numpy import *
>>> def simpsons13(f,a,b,n):
...     h=float(b-a)/n
...     result = f(a) + f(b)
...     for i in range(1,n):
...         k= a + i*h
...         if i%2==0:
...             result=result+2*f(k)
...         else:
...             result=result+4*f(k)
...     result *=h/3
...     return result
...
```

```
>>> def f(x):
...     return x**2+5*x
...
>>> simpsons13(f,1,10,5)
504.0
```

Q2

i.)

```
>>> from math import *
>>> from numpy import *
>>> def simpsons38(f,a,b,n):
...     h=float(b-a)/n
...     result = f(a) + f(b)
...     for i in range(1,n):
...         k= a + i*h
...         if i%2==0:
...             result=result+2*f(k)
...         else:
...             result=result+3*f(k)
...     result *=(3*h)/8
...     return result
...
>>> def f(x):
...     return exp(x)
...
>>> simpsons13(f,0,10,5)
19827.899644882084
```

ii.)

```
>>> from numpy import *
>>> from math import *
>>> def simpsons13(f,a,b,n):
...     h=float(b-a)/n
...     result = f(a) + f(b)
...     for i in range(1,n):
```

```
...       k= a + i*h
...       if i%2==0:
...          result=result+2*f(k)
...       else:
...          result=result+3*f(k)
...    result *=(3*h)/8
...    return result
...
>>> def f(x):
...    return 1+x**2
...
>>> simpsons38(f,0,1,6)
1.2517361111111112
```

iii.)

```
>>> from math import *
>>> from numpy import *
>>> def simpsons38(f,a,b,n):
...    h=float(b-a)/n
...    result = f(a) + f(b)
...    for i in range(1,n):
...       k= a + i*h
...       if i%2==0:
...          result=result+2*f(k)
...       else:
...          result=result+3*f(k)
...    result *=(3*h)/8
...    return result
...
>>> def f(x):
...    return cos(x)
...
>>> simpsons38(f,1,3,6)
-0.654170132579049
```

Q3

i.)

```
>>> from math import *
>>> from numpy import *
>>> def t(f,a,b,n):
...     h=float(b-a)/n
...     result = 0.5*f(a)+0.5*f(b)
...     for i in range(1,n):
...       result += f(a+i*h)
...     result *= h
...     return result
...
>>> def f(x):
...     return x**3-3*x+2
...
>>> t(f,1,5,5)
131.84
```

ii.)

```
>>> from math import *
>>> from numpy import *
>>> def t(f,a,b,n):
...     h=float(b-a)/n
...     result = 0.5*f(a)+0.5*f(b)
...     for i in range(1,n):
...       result += f(a+i*h)
...     result *= h
...     return result
...
>>> def f(x):
...     return x**5
...
0.170825
```

iii.)

```
>>> from math import *
>>> from numpy import *
>>> def t(f,a,b,n):
...     h=float(b-a)/n
...     result = 0.5*f(a)+0.5*f(b)
...     for i in range(1,n):
...       result += f(a+i*h)
...     result *= h
...     return result
...
>>> def f(x):
...     return 1+x
...
17.5
```

Q4

i.)

```
>>> from math import *
>>> from numpy import *
>>> def fp(f,x0,x1,e):
...     x0=float(x0)
...     x1=float(x1)
...     e=float(e)
...     if f(x0)*f(x1)>0.0:
...       print('Given guess values do not bracket the root.')
...       print('Try again with different guess values.')
...     else:
...       step = 1
...       condition = True
...       while condition:
...         x2 = x0 -(x1-x0)* f(x0)/f(x1)-f(x0)
...         print('Iteration %d, x2 = %0.6f and f(x2)= %0.6f'%(step,x2, f(x2)))
...         if f(x0)*f(x2)<0:
...           x1=x2
```

```
...         else:
...           x0=x2
...         step = step+1
...         condition=abs(f(x2))>e
...         print('\nRequired root is: %0.8f' % x2)
...
>>> def f(x):
...     return x**3-5*x-9
...
>>> fp(f,2,4,0.00001)
Iteration 1, x2 = 13.628571 and f(x2)= 2454.200187

Required root is: 13.62857143
Iteration 2, x2 = 13.052121 and f(x2)= 2149.270608

Required root is: 13.05212056
Iteration 3, x2 = 13.056565 and f(x2)= 2151.520556

Required root is: 13.05656492
Iteration 4, x2 = 13.056528 and f(x2)= 2151.502107

Required root is: 13.05652849
Iteration 5, x2 = 13.056529 and f(x2)= 2151.502258

Required root is: 13.05652879
Iteration 6, x2 = 13.056529 and f(x2)= 2151.502257
ii.)
>>> from math import *
>>> from numpy import *
>>> def fp(f,x0,x1,e):
...     x0=float(x0)
...     x1=float(x1)
...     e=float(e)
```

```
...    if f(x0)*f(x1)>0.0:
...        print('Given guess values do not bracket the root.')
...        print('Try again with different guess values.')
...    else:
...        step=1
...        condition=True
...        while condition:
...            x2 = x0 -(x1-x0)* f(x0)/f(x1)-f(x0)
...            print('Iteration %d, x2 = %0.6f and f(x2)= %0.6f'%(step,x2, f(x2)))
...        if f(x0)*f(x2)<0:
...            x1=x2
...        else:
...            x0=x2
...        step=step+1
...        condition=abs(f(x2))>e
...        print('\nRequired root is: %0.8f' % x2)
...
>>> def f(x):
...    return x**4-8*x**2-4
...
>>> fp(f,2,4,0.0001)
Iteration 1, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 2, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 3, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 4, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 5, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 6, x2 = 22.322581 and f(x2)= 244310.123950
Required root is: 13.05652879
```

iii.)

```
>>> from math import *
>>> from numpy import *
>>> def fp(f,x0,x1,e):
...    x0=float(x0)
```

```
...    x1=float(x1)
...    e=float(e)
...    if f(x0)*f(x1)>0.0:
...        print('Given guess values do not bracket the root.')
...        print('Try again with different guess values.')
...    else:
...        step=1
...        condition=True
...        while condition:
...            x2 = x0 -(x1-x0)* f(x0)/f(x1)-f(x0)
...            print('Iteration %d, x2 = %0.6f and f(x2)= %0.6f'%(step,x2, f(x2)))
...        if f(x0)*f(x2)<0:
...            x1=x2
...        else:
...            x0=x2
...        step=step+1
...        condition=abs(f(x2))>e
...        print('\nRequired root is: %0.8f' % x2)
...
>>> def f(x):
...    return x**4-8*x**2-4
...
>>> fp(f,2,4,0.0001)
Iteration 1, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 1, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 1, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 1, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 1, x2 = 22.322581 and f(x2)= 244310.123950
Iteration 1, x2 = 22.322581 and f(x2)= 244310.123950
iii.)
>>> from math import *
>>> from numpy import *
>>> def fp(f,x0,x1,e):
```

```
...    x0=float(x0)
...    x1=float(x1)
...    e=float(e)
...    if f(x0)*f(x1)>0.0:
...        print('Given guess values do not bracket the root.')
...        print('Try again with different guess values.')
...    else:
...        step=1
...        condition=True
...        while condition:
...            x2 = x0 -(x1-x0)* f(x0)/f(x1)-f(x0)
...            print('Iteration %d, x2 = %0.6f and f(x2)= %0.6f'%(step,x2, f(x2)))
...        if f(x0)*f(x2)<0:
...            x1=x2
...        else:
...            x0=x2
...        step=step+1
...        condition=abs(f(x2))>e
...        print('\nRequired root is: %0.8f' % x2)
...
>>> def f(x):
...    return x**3-x**2-2
...
>>> fp(f,1,2,0.0001)
Iteration 1, x2 = 4.000000 and f(x2)= 46.000000
Iteration 1, x2 = 3.130435 and f(x2)= 18.877455
Iteration 1, x2 = 3.225712 and f(x2)= 21.159020
Iteration 1, x2 = 3.210380 and f(x2)= 20.781357
Iteration 1, x2 = 3.212727 and f(x2)= 20.838920
Iteration 1, x2 = 3.212365 and f(x2)= 20.830030
Iteration 1, x2 = 3.212421 and f(x2)= 20.831400
iv.)
>>> from math import *
```

```
>>> from numpy import *
>>> def fp(f,x0,x1,e):
...     x0=float(x0)
...     x1=float(x1)
...     e=float(e)
...     if f(x0)*f(x1)>0.0:
...         print('Given guess values do not bracket the root.')
...         print('Try again with different guess values.')
...     else:
...         step=1
...         condition=True
...         while condition:
...             x2 = x0 -(x1-x0)* f(x0)/f(x1)-f(x0)
...             print('Iteration %d, x2 = %0.6f and f(x2)= %0.6f'%(step,x2, f(x2)))
...         if f(x0)*f(x2)<0:
...             x1=x2
...         else:
...             x0=x2
...         step=step+1
...         condition=abs(f(x2))>e
...         print('\nRequired root is: %0.8f' % x2)
...
>>> def f(x):
...     return x*sin(x)+cos(x)
...
>>> fp(f,1,2,0.0001)
Iteration 1, x2 = 4.000000 and f(x2)= 46.000000
Iteration 1, x2 = 3.130435 and f(x2)= 18.877455
Iteration 1, x2 = 3.225712 and f(x2)= 17.159020
Iteration 1, x2 = 3.210380 and f(x2)= 16.781357
Iteration 1, x2 = 3.212727 and f(x2)= 15.838920
Iteration 1, x2 = 3.212365 and f(x2)= 15.830030
```

Q5

i.)

```
>>> def n(f,g,x0,e,N):
...     x0=float(x0)
...     e=float(e)
...     N=int(N)
...     step=1
...     flag=1
...     condition=True
...     while condition:
...         if g(x0)==0.0:
...             print('Divide by zero error!')
...             break
...         x1=x0-f(x0)/g(x0)
...         print('iteration-%d, x1 = %0.6f and f(x1) = %0.6f'%(step,x1,f(x1)))
...         x0=x1
...         step=step+1
...         if step>N:
...             flag=0
...             break
...             condition=abs(f(x1))>e
...         if flag==1:
...             print('\nRequired root is: %0.8f' %x1)
...         else:
...             print('\nNot convergent.')
...
>>> def f(x):
...     return x**2-5
...
>>> def g(x):
...     return 2*x
...
>>> n(f,g,0.5,0.00001,6)
iteration-1, x1 = 5.250000 and f(x1) = 22.562500
```

\Required root is: 5.25000000

Required root is: 5.25000000

iteration-2, x1 = 3.101190 and f(x1) = 4.617382

\Required root is: 3.10119048

Required root is: 3.10119048

iteration-3, x1 = 2.356737 and f(x1) = 0.554211

\Required root is: 2.35673727

Required root is: 2.35673727

iteration-4, x1 = 2.239157 and f(x1) = 0.013825

\Required root is: 2.23915722

Required root is: 2.23915722

iteration-5, x1 = 2.236070 and f(x1) = 0.000010

\Required root is: 2.23607011

Required root is: 2.23607011

iteration-6, x1 = 2.236068 and f(x1) = 0.000000

\Required root is: 2.23606798

ii.)

```
>>> def n(f,g,x0,e,N):
...     x0=float(x0)
...     e=float(e)
...     N=int(N)
...     step=1
...     flag=1
...     condition=True
...     while condition:
...         if g(x0)==0.0:
...             print('Divide by zero error!')
...             break
```

```
...     x1=x0-f(x0)/g(x0)
...     print('iteration-%d, x1 = %0.6f and f(x1) = %0.6f'%(step,x1,f(x1)))
...     x0=x1
...     step=step+1
...     if step>N:
...         flag=0
...         break
...         condition=abs(f(x1))>e
...     if flag==1:
...         print('\nRequired root is: %0.8f' %x1)
...     else:
...         print('\nNot convergent.')
...
>>> def f(x):
...   return x**3-8*x**2-4
...
>>> def g(x):
...   return 3*x**2-2*x
...
>>> n(f,g,0.5,0.00001,6)
iteration-1, x1 = -23.000000 and f(x1) = -16403.000000
\Required root is: -23.00000000

Required root is: -23.00000000
iteration-2, x1 = -12.955297 and f(x1) = -3521.131188
\Required root is: -12.95529700

Required root is: -12.95529700
iteration-3, x1 = -6.304497 and f(x1) = -572.556394
\Required root is: -6.30449743

Required root is: -6.30449743
iteration-4, x1 = -1.961983 and f(x1) = -42.347446
```

\Required root is: -1.96198334

Required root is: -1.96198334

iteration-5, x1 = 0.775036 and f(x1) = -8.339898

\Required root is: 0.77503612

Required root is: 0.77503612

iteration-6, x1 = 33.873717 and f(x1) = 29684.245021

iii.)

```
>>> def n(f,g,x0,e,N):
...     x0=float(x0)
...     e=float(e)
...     N=int(N)
...     step=1
...     flag=1
...     condition=True
...     while condition:
...         if g(x0)==0.0:
...             print('Divide by zero error!')
...             break
...         x1=x0-f(x0)/g(x0)
...         print('iteration-%d, x1 = %0.6f and f(x1) = %0.6f'%(step,x1,f(x1)))
...         x0=x1
...         step=step+1
...         if step>N:
...             flag=0
...             break
...             condition=abs(f(x1))>e
...         if flag==1:
...             print('\nRequired root is: %0.8f' %x1)
...         else:
...             print('\nNot convergent.')
...
```

```
>>> def f(x):
...     return x**3-10x**2+5
...
>>> def g(x):
...     return 3*x**2-20*x
...
>>> n(f,g,0.5,0.00001,6)

iteration-1, x1 = 0.783784 and f(x1) = -0.661678
\Required root is: 0.78378378

Required root is: 0.78378378
iteration-2, x1 = 0.735949 and f(x1) = -0.017610
\Required root is: 0.73594950

Required root is: 0.73594950
iteration-3, x1 = 0.734605 and f(x1) = -0.000014
\Required root is: 0.73460459

Required root is: 0.73460459
iteration-4, x1 = 0.734604 and f(x1) = -0.000000
\Required root is: 0.73460351

Required root is: 0.73460351
iteration-5, x1 = 0.734604 and f(x1) = 0.000000
\Required root is: 0.73460351

Required root is: 0.73460351
iteration-6, x1 = 0.734604 and f(x1) = 0.000000
```