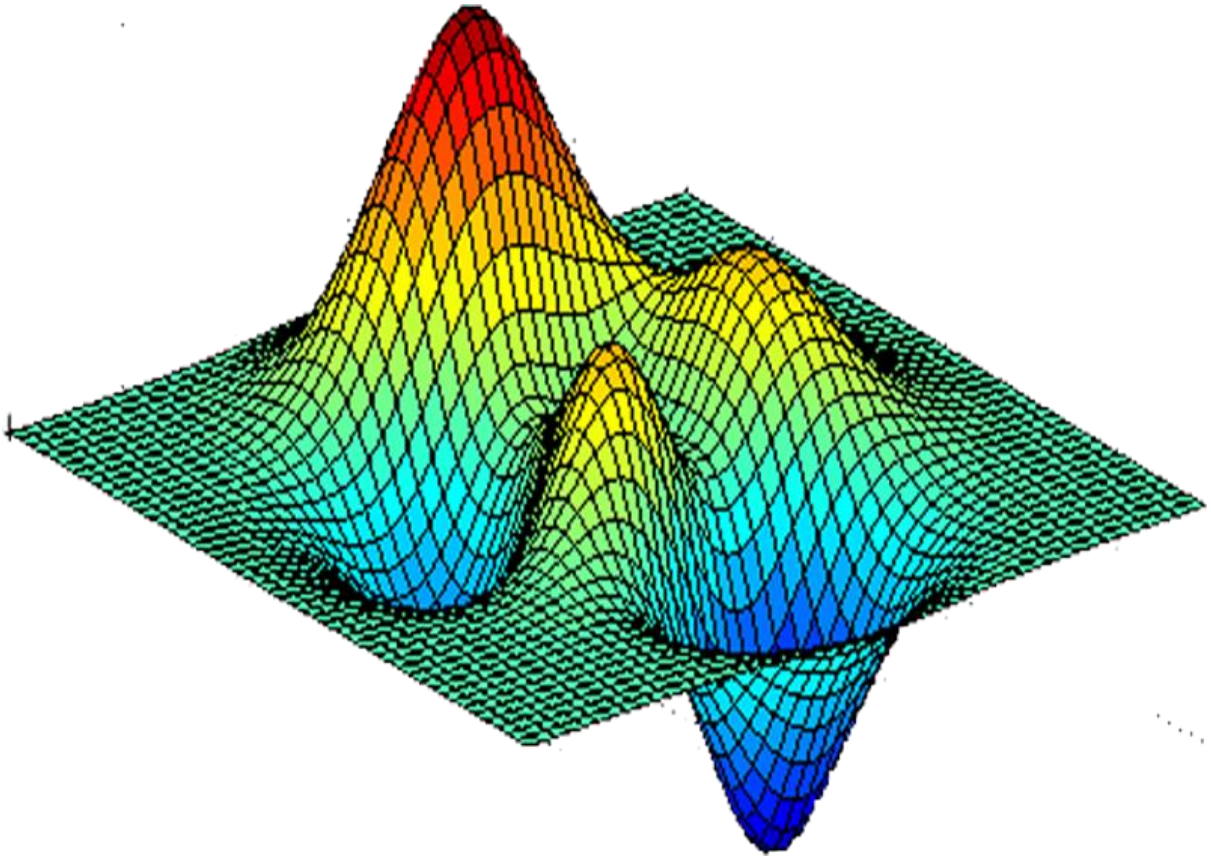
Numerical Computing

Muller's Method and The Quotient Difference Algorithm



QASID AHMED ALEEM

COURSE TEACHER: DR. SHAHID QURESHI
BSC(HONOURS)
H-2 (MORNING)
INSTITUTE OF SPACE AND PLANETARY ASTROPHYSICS
UNIVERSITY OF KARACHI

Tasks:

Task1: Muller's Method

Code:

```
from math import sin,exp

def func(f):
        return (3*f) + sin(f) - exp(f)
x0=float(input("Enter the first approximation of x:"))
x1=float(input("Enter the second approximation of x: "))
x2=float(input("Enter the third approximation of x: "))

approx1=[x0,x1,x2]
approx1.sort()
x4=0
def a1(g,f0,f1,f2,h1):
        return ((g*f2)+(f0)-(f1*(1+g)))/((g*(h1**2))*(1+g))

def b1(f1,f2,h1,a1):
        return (f2-f1-(a1*(h1**2)))/h1

def x_newm(a,b,c,x1):
        return x1-((2*c)/(b -(((b**2)-(4*a*c))**0.5)))
def x_newp(a,b,c,x1):
        return x1-((2*c)/(b +(((b**2)-(4*a*c))**0.5)))

print("{:^13s} {:^13s} {:^13s} {:^13s} {:^13s} {:^13s} {:^13s} {:^13s} \
{:^13s} {:^13s} {:^13s} {:^13s}".format("x2","x0","x1","f2","f0","f1",\
"h1","h2","gamma","a","b","x_new"))
for i in range(0,100):
        h1=approx1[2]-approx1[1]
        h2=approx1[1]-approx1[0]

        g=h2/h1

        a=a1(g,func(approx1[0]),func(approx1[1]),func(approx1[2]),h1)
        b=b1(func(approx1[1]),func(approx1[2]),h1,a)
        c=func(approx1[1])

        if b>0:
                x3=x_newp(a,b,c,approx1[1])
        else:
                x3=x_newm(a,b,c,approx1[1])

        if x4==x3:
                break
        #print(x3)
        approx1.append(x3)
        approx1.sort()
        del approx1[3]
        #print (approx1)
        x4=x3
        print("{:13.10f} {: 13.10f} {: 13.10f} {: 13.10f} {: 13.10f} \
{: 13.10f} {: 13.10f} {: 13.10f} {: 14.10f} {: 13.10f} {: 13.10f} \
{: 13.10f}".format(approx1[0],approx1[1],approx1[2],func(approx1[0]),\
func(approx1[1]),func(approx1[2]),h1,h2,g,a,b,x3))
```

```
        #print(x4)

print("the approximate root after {:1d} iterations is {:36.34f} ".\
format(i,x3))
```

Output:

```
Enter the first approximation of x:0.5
Enter the second approximation of x: 0
Enter the third approximation of x: 1
     x2           x0           x1           f2           f0           f1           h1           h2          gamma          a            b          x_new
 0.0000000000  0.3549138905  0.5000000000 -1.0000000000 -0.0138065567  0.3307042679  0.5000000000  0.5000000000   1.0000000000 -1.0764387589  2.1231891563  0.3549138905

 0.0000000000  0.3549138905  0.3604646779 -1.0000000000 -0.0138065567  0.0001075137  0.1450861095  0.3549138905   2.4462292889 -0.8083142284  2.4918017549  0.3604646779

 0.0000000000  0.3549138905  0.3604217159 -1.0000000000 -0.0138065567  0.0000000323  0.0055507874  0.3549138905  63.9393769766 -0.7545805422  2.5108725865  0.3604217159

 0.0000000000  0.3549138905  0.3604217030 -1.0000000000 -0.0138065567  0.0000000000  0.0055078254  0.3549138905  64.4381158754 -0.7545642639  2.5108783639  0.3604217030

 0.0000000000  0.3549138905  0.3604217030 -1.0000000000 -0.0138065567  0.0000000000  0.0055078125  0.3549138905  64.4382669457 -0.7545642590  2.5108783657  0.3604217030

 0.0000000000  0.3549138905  0.3604217030 -1.0000000000 -0.0138065567  0.0000000000  0.0055078125  0.3549138905  64.4382669912 -0.7545642590  2.5108783657  0.3604217030

 0.0000000000  0.3549138905  0.3604217030 -1.0000000000 -0.0138065567  0.0000000000  0.0055078125  0.3549138905  64.4382669912 -0.7545642590  2.5108783657  0.3604217030

the approximate root after 7 iterations is 0.3604217029603243860336192483373452
Press any key to continue . . .
```

Spreadsheet work:

x2	x0	x1	f2	f0	f1	h1	h2	g	a	b	x new
0.000000	0.500000	1.000000	-1.000000	0.330704	1.123189	0.500000	0.500000	1.000000	-1.076439	2.123189	**0.35491389049**
0.000000	0.354914	0.500000	-1.000000	-0.013807	0.330704	0.145086	0.354914	2.446229	-0.808314	2.491802	**0.36046467793**
0.000000	0.360465	0.500000	-1.000000	0.000108	0.330704	0.139535	0.360465	2.583322	-0.810452	2.482356	**0.36042136737**
0.000000	0.360421	0.500000	-1.000000	-0.000001	0.330704	0.139579	0.360421	2.582210	-0.810435	2.482430	**0.36042170558**
0.000000	0.360422	0.500000	-1.000000	0.000000	0.330704	0.139578	0.360422	2.582219	-0.810435	2.482429	**0.36042170294**
0.000000	0.360422	0.500000	-1.000000	0.000000	0.330704	0.139578	0.360422	2.582219	-0.810435	2.482429	**0.36042170296**
0.000000	0.360422	0.500000	-1.000000	0.000000	0.330704	0.139578	0.360422	2.582219	-0.810435	2.482429	**0.36042170296**
0.000000	0.360422	0.500000	-1.000000	0.000000	0.330704	0.139578	0.360422	2.582219	-0.810435	2.482429	**0.36042170296**
0.000000	0.360422	0.500000	-1.000000	0.000000	0.330704	0.139578	0.360422	2.582219	-0.810435	2.482429	**0.36042170296**
0.000000	0.360422	0.500000	-1.000000	0.000000	0.330704	0.139578	0.360422	2.582219	-0.810435	2.482429	**0.36042170296**
0.000000	0.360422	0.500000	-1.000000	0.000000	0.330704	0.139578	0.360422	2.582219	-0.810435	2.482429	**0.36042170296**
0.000000	0.360422	0.500000	-1.000000	0.000000	0.330704	0.139578	0.360422	2.582219	-0.810435	2.482429	**0.36042170296**

Task 2: QD ALGORITHM

Code:

```
quest=int(input("what is the highest power of x: "))
coefficents=[]
qs=[]
es=[]
qsn=[]
esn=[]
for i in range(0,quest+1):

        coefficent=float(input("Enter the coefficent of x^{}: ".format(i)))
        coefficents.append(coefficent)
coefficents.reverse()
d=int(input("How many iterations are required: "))
for s in range(0,quest+1):
        print("{:^14s}".format("e"+str(s),"q"+str(s)),end=" ")
        if s==4:
                break
```

```
            print("{:^14s}".format("q"+str(s)),end=" ")
print(" ")
for i in range(0,quest):
            if i==0:
                        qsi=(-coefficents[1]/coefficents[0])
                        qs.append(qsi)
            else:
                        qsi=0
                        qs.append(qsi)
for s in range(0,quest):
            print("{:15s}{: 14.12f}".format(" ",qs[s]),end=" ")
print(" ")


for i in range(0,quest+1):
            if i==0 or i==quest:
                        esi=0
                        es.append(esi)
            else:
                        esi=coefficents[i+1]/coefficents[i]
                        es.append(esi)
for s in range(0,quest+1):
            print("{: 14.12f}{:15s}".format(es[s]," "),end=" ")
print(" ")

for i in range(0,d):
            for s in range(0,quest):
                        qsi=es[s+1]-es[s] +qs[s]
                        if i ==0:
                                    qsn.append(qsi)
                        else:
                                    qsn[s]=qsi
                                    #qs=qsn
            for s in range(0,quest+1):
                        if s==0 or s==quest:
                                    esi=0
                                    if i==0:
                                                esn.append(esi)
                                    else:
                                                esn[s]=esi

                        else:
                                    esi=(qsn[s]/qsn[s-1])*(es[s])
                                    if i==0:
                                                esn.append(esi)
                                    else:
                                                esn[s]=esi

            es=esn
            qs=qsn
            for s in range(0,quest):
                        print("{:15s}{: 14.12f}".format(" ",qsn[s]),end=" ")
            print(" ")

            for s in range(0,quest+1):
                        print("{: 14.12f}{:15s}".format(esn[s]," "),end=" ")
```

```
73            print(" ")
74 print("The {:2d} approximate roots of the function are".format(quest))
75 for s in range(0,quest):
76            print("{: 18.16}".format(qsn[s]))
```

Output:

```
what is the highest power of x: 4
Enter the coefficent of x^0: 1
Enter the coefficent of x^1: -32
Enter the coefficent of x^2: 160
Enter the coefficent of x^3: -256
Enter the coefficent of x^4: 128
How many iterations are required: 15
      e0             q0             e1             q1             e2             q2             e3             q3             e4
                2.000000000000                0.000000000000                0.000000000000                0.000000000000
 0.000000000000               -0.625000000000               -0.200000000000               -0.031250000000                0.000000000000
                1.375000000000                0.425000000000                0.168750000000                0.031250000000
 0.000000000000               -0.193181818182               -0.079411764706               -0.005787037037                0.000000000000
                1.181818181818                0.538770053476                0.242374727669                0.037037037037
 0.000000000000               -0.088068181818               -0.035724711721               -0.000884311278                0.000000000000
                1.093750000000                0.591113523573                0.277215128112                0.037921348315
 0.000000000000               -0.047596153846               -0.016753855464               -0.000120968420                0.000000000000
                1.046153846154                0.621955821956                0.293848015156                0.038042316734
 0.000000000000               -0.028296703297               -0.007915493352               -0.000015660881                0.000000000000
                1.017857142857                0.642337031900                0.301747847627                0.038057977616
 0.000000000000               -0.017857142857               -0.003718426563               -0.000001975230                0.000000000000
                1.000000000000                0.656475748194                0.305464298960                0.038059952846
 0.000000000000               -0.011722781218               -0.001730218620               -0.000000246108                0.000000000000
                0.988277218782                0.666468310791                0.307194271472                0.038060198954
 0.000000000000               -0.007905537078               -0.000797507158               -0.000000030492                0.000000000000
                0.980371681705                0.673576340711                0.307991748139                0.038060229446
 0.000000000000               -0.005431595828               -0.000364658924               -0.000000003768                0.000000000000
                0.974940085877                0.678643277614                0.308356403295                0.038060233214
 0.000000000000               -0.003780864126               -0.000165690751               -0.000000000465                0.000000000000
                0.971159221751                0.682258450989                0.308522093581                0.038060233679
 0.000000000000               -0.002656131398               -0.000074926529               -0.000000000057                0.000000000000
                0.968503090353                0.684839655858                0.308597020052                0.038060233736
 0.000000000000               -0.001878181010               -0.000033762799               -0.000000000007                0.000000000000
                0.966624909343                0.686684074070                0.308630782844                0.038060233743
 0.000000000000               -0.001334247623               -0.000015174721               -0.000000000001                0.000000000000
                0.965290661719                0.688003146973                0.308645957564                0.038060233744
 0.000000000000               -0.000950974251               -0.000006807551               -0.000000000000                0.000000000000
                0.964339687469                0.688947313673                0.308652765114                0.038060233744
 0.000000000000               -0.000679398726               -0.000003049826               -0.000000000000                0.000000000000
                0.963660288742                0.689623662574                0.308655814940                0.038060233744
 0.000000000000               -0.000486197723               -0.000001365015               -0.000000000000                0.000000000000
The  4 approximate roots of the function are
 0.9636602887421978
 0.6896236625735249
 0.3086558149399226
 0.03806023374435475
Press any key to continue . . .
```

Spreadsheet work:

e0	q0	e1	q1	e2	q2	e3	q3	e4
	2		0		0		0	
0		-0.625		-0.2		-0.03125		0
	1.375		0.425		0.16875		0.03125	
0		-0.19318		-0.07941		-0.00579		0
	1.1818182		0.53877		0.242375		0.037037	
0		-0.08807		-0.03572		-0.00088		0
	1.093750		0.591114		0.277215		0.037921	
0		-0.0476		-0.01675		-0.00012		0
	1.046154		0.621956		0.293848		0.038042	
0		-0.0283		-0.00792		-1.6E-05		0
	1.017857		0.642337		0.301748		0.038058	
0		-0.01786		-0.00372		-2E-06		0
	1.000000		0.656476		0.305464		0.03806	
0		-0.01172		-0.00173		-2.5E-07		0
	0.988277		0.666468		0.307194		0.03806	
0		-0.00791		-0.0008		-3E-08		0
	0.980372		0.673576		0.307992		0.03806	
0		-0.00543		-0.00036		-3.8E-09		0
	0.974940		0.678643		0.308356		0.03806	
0		-0.00378		-0.00017		-4.7E-10		0
	0.971159		0.682258		0.308522		0.03806	
0		-0.00266		-7.5E-05		-5.7E-11		0
	0.968503		0.68484		0.308597		0.03806	
0		-0.00188		-3.4E-05		-7.1E-12		0
	0.966625		0.686684		0.308631		0.03806	
0		-0.00133		-1.5E-05		-8.7E-13		0