Labratory 6 Submission

MA 202 Numerical Techniques (2021-22)

Name - Rajan Kumar

Roll No - 202051152

METHODS FOR FINDING ROOTS :

- ***CLOSED METHODS :***

1. ***BISECTION METHOD.***
2. ***FALSE POSITION.***

- ***OPEN METHODS :***

1. ***NEWTON-RAPHSON'S METHOD.***
2. ***SECANT METHOD.***

Q1.

A) Write routine in MatLab for finding the root of any given general function for each of the methods that we have discussed. The program should not only provide the root with the uncertainty/error, but also provide

the user with the no. of iterations used to reach the root, which is indicative of the rate of convergence.

B) Use these routines to find roots of these two functions f = ln(x) - 1 and g = tanh(x).

```
syms x
f = @(x)log(x) - 1;
Bisection(f,2,3,1e-4)
```

```
-------------------------------------------------------
Root using Bisection Method :
n = 2.7183
itr = 11
-------------------------------------------------------
```

```
FalsePosition(f,-1,3,1e-4)
```

```
1)    x1=2.960877
2)    x2=2.883582
3)    x3=2.786991
4)    x4=2.689520
5)    x5=2.728119
6)    x6=2.718249
```

```
7)    x7=2.718282
-------------------------------------------------------


Root using False Position Method : -------------------------------------------------------
```

```
N_R(f,10,3,1e-4,x)
```

```
step=1    a=3.000000    f(a)=0.098612
step=2    a=2.704163    f(a)=-0.005208
step=3    a=2.718245    f(a)=-0.000014
-------------------------------------------------------

Root using Newton-Raphsonps Method :
Root is 2.718282
-------------------------------------------------------
```

```
Secant(f,-1,3,1e-4)
```

```
   Xn-1      f(Xn-1)      Xn      f(Xn)      Xn+1      f(Xn+1)
  Column 1

  -1.0000 + 0.0000i

  Column 2

  -1.0000 + 3.1416i

  Column 3

   3.0000 + 0.0000i

  Column 4

   0.0986 + 0.0000i

  Column 5

   2.9609 - 0.1119i

  Column 6

   0.0862 - 0.0378i

  Column 1

   3.0000 + 0.0000i

  Column 2

   0.0986 + 0.0000i

  Column 3

   2.9609 - 0.1119i

  Column 4

   0.0862 - 0.0378i

  Column 5

   2.7061 + 0.0055i
```

```
  Column 6

  -0.0045 + 0.0020i

  Column 1

   2.9609 - 0.1119i

  Column 2

   0.0862 - 0.0378i

  Column 3

   2.7061 + 0.0055i

  Column 4

  -0.0045 + 0.0020i

  Column 5

   2.7187 - 0.0005i

  Column 6

   0.0002 - 0.0002i

  Column 1

   2.7061 + 0.0055i

  Column 2

  -0.0045 + 0.0020i

  Column 3

   2.7187 - 0.0005i

  Column 4

   0.0002 - 0.0002i

  Column 5

   2.7183 - 0.0000i

  Column 6

   0.0000 - 0.0000i

--------------------------------------------------------

Root using Secant Method :
Root is x = 2.7183-1.5377e-06i
--------------------------------------------------------
```

```
g = @(x)tanh(x);
Bisection(g,-1,1,1e-4)
```

```
--------------------------------------------------------
Root using Bisection Method :
n = 0
itr = 0
--------------------------------------------------------
```

```
FalsePosition(g,-1,3,1e-4)
```

```
1)    x1=0.734198
2)    x2=-0.047913
3)    x3=0.007684
4)    x4=-0.000005
--------------------------------------------------------


Root using False Position Method : --------------------------------------------------------
```

```
N_R(g,10,-1,1e-4,x)
```

```
step=1    a=-1.000000    f(a)=-0.761594
step=2    a=0.813430    f(a)=0.671478
step=3    a=-0.409402    f(a)=-0.387965
step=4    a=0.047305    f(a)=0.047270
step=5    a=-0.000071    f(a)=-0.000071
--------------------------------------------------------

Root using Newton-Raphsonps Method :
Root is 0.000000
--------------------------------------------------------
```

```
Secant(g,-2,-1,1e-4)
```

```
   Xn-1      f(Xn-1)      Xn      f(Xn)      Xn+1      f(Xn+1)
  Column 1

   -2.0000

  Column 2

   -0.9640

  Column 3

   -1.0000

  Column 4

   -0.7616

  Column 5

    2.7622

  Column 6

    0.9921

  Column 1

   -1.0000

  Column 2
```

```
 -0.7616
Column 3
  2.7622
Column 4
  0.9921
Column 5
  0.6339
Column 6
  0.5607
Column 1
  2.7622
Column 2
  0.9921
Column 3
  0.6339
Column 4
  0.5607
Column 5
 -2.1329
Column 6
 -0.9723
Column 1
  0.6339
Column 2
  0.5607
Column 3
 -2.1329
Column 4
 -0.9723
Column 5
 -0.3781
Column 6
```

```
 -0.3610
Column 1
 -2.1329
Column 2
 -0.9723
Column 3
 -0.3781
Column 4
 -0.3610
Column 5
  0.6584
Column 6
  0.5773
Column 1
 -0.3781
Column 2
 -0.3610
Column 3
  0.6584
Column 4
  0.5773
Column 5
  0.0207
Column 6
  0.0207
Column 1
  0.6584
Column 2
  0.5773
Column 3
  0.0207
Column 4
```

```
    0.0207

  Column 5

   -0.0030

  Column 6

   -0.0030

  Column 1

    0.0207

  Column 2

    0.0207

  Column 3

   -0.0030

  Column 4

   -0.0030

  Column 5

    0.0000

  Column 6

    0.0000

--------------------------------------------------------

Root using Secant Method :
Root is x = 3.6873e-07
--------------------------------------------------------
```

Q2.

Use Newton-Raphson's method to find the root of the function f(x) = sech(x), using the initial guess as x = 0.

Write in detail the output that you obtain and comment upon it.

```
syms x
f = @(x)sech(x);
N_R(f,10,1,1e-4,x)
```

```
step=1    a=1.000000    f(a)=0.648054
step=2    a=2.313035    f(a)=0.196001
step=3    a=3.332815    f(a)=0.071294
step=4    a=4.335366    f(a)=0.026190
step=5    a=5.335710    f(a)=0.009633
step=6    a=6.335756    f(a)=0.003544
step=7    a=7.335762    f(a)=0.001304
step=8    a=8.335763    f(a)=0.000480
step=9    a=9.335763    f(a)=0.000176
```

```
step=10    a=10.335763    f(a)=0.000065
-------------------------------------------------------

Root using Newton-Raphsonps Method :
Root is 11.335763
-------------------------------------------------------
```

Q3.

Find the root of the function f(x) = 1/x, using all the four methods. Please comment upon the answer you find using each method.

```
syms x
f = @(x)1 / x;
Bisection(f,2,30,1e-4)
```

```
-------------------------------------------------------
Root using Bisection Method :
n = 0
itr = 0
-------------------------------------------------------
```

```
FalsePosition(f,1,30,1e-4)
```

```
root not located between the entered values
1)    x1=31.000000
2)    x2=61.000000
3)    x3=91.000000
4)    x4=121.000000
5)    x5=151.000000
6)    x6=181.000000
7)    x7=211.000000
8)    x8=241.000000
9)    x9=271.000000
10)    x10=301.000000
11)    x11=331.000000
12)    x12=361.000000
13)    x13=391.000000
14)    x14=421.000000
15)    x15=451.000000
16)    x16=481.000000
17)    x17=511.000000
18)    x18=541.000000
19)    x19=571.000000
20)    x20=601.000000
21)    x21=631.000000
22)    x22=661.000000
23)    x23=691.000000
24)    x24=721.000000
25)    x25=751.000000
26)    x26=781.000000
27)    x27=811.000000
28)    x28=841.000000
29)    x29=871.000000
30)    x30=901.000000
31)    x31=931.000000
32)    x32=961.000000
33)    x33=991.000000
34)    x34=1021.000000
35)    x35=1051.000000
```

```
36)    x36=1081.000000
37)    x37=1111.000000
38)    x38=1141.000000
39)    x39=1171.000000
40)    x40=1201.000000
41)    x41=1231.000000
42)    x42=1261.000000
43)    x43=1291.000000
44)    x44=1321.000000
45)    x45=1351.000000
46)    x46=1381.000000
47)    x47=1411.000000
48)    x48=1441.000000
49)    x49=1471.000000
50)    x50=1501.000000
51)    x51=1531.000000
52)    x52=1561.000000
53)    x53=1591.000000
54)    x54=1621.000000
55)    x55=1651.000000
56)    x56=1681.000000
57)    x57=1711.000000
58)    x58=1741.000000
59)    x59=1771.000000
60)    x60=1801.000000
61)    x61=1831.000000
62)    x62=1861.000000
63)    x63=1891.000000
64)    x64=1921.000000
65)    x65=1951.000000
66)    x66=1981.000000
67)    x67=2011.000000
68)    x68=2041.000000
69)    x69=2071.000000
70)    x70=2101.000000
71)    x71=2131.000000
72)    x72=2161.000000
73)    x73=2191.000000
74)    x74=2221.000000
75)    x75=2251.000000
76)    x76=2281.000000
77)    x77=2311.000000
78)    x78=2341.000000
79)    x79=2371.000000
80)    x80=2401.000000
81)    x81=2431.000000
82)    x82=2461.000000
83)    x83=2491.000000
84)    x84=2521.000000
85)    x85=2551.000000
86)    x86=2581.000000
87)    x87=2611.000000
88)    x88=2641.000000
89)    x89=2671.000000
90)    x90=2701.000000
91)    x91=2731.000000
92)    x92=2761.000000
93)    x93=2791.000000
94)    x94=2821.000000
95)    x95=2851.000000
96)    x96=2881.000000
97)    x97=2911.000000
98)    x98=2941.000000
99)    x99=2971.000000
```

```
100)    x100=3001.000000
101)    x101=3031.000000
102)    x102=3061.000000
103)    x103=3091.000000
104)    x104=3121.000000
105)    x105=3151.000000
106)    x106=3181.000000
107)    x107=3211.000000
108)    x108=3241.000000
109)    x109=3271.000000
110)    x110=3301.000000
111)    x111=3331.000000
112)    x112=3361.000000
113)    x113=3391.000000
114)    x114=3421.000000
115)    x115=3451.000000
116)    x116=3481.000000
117)    x117=3511.000000
118)    x118=3541.000000
119)    x119=3571.000000
120)    x120=3601.000000
121)    x121=3631.000000
122)    x122=3661.000000
123)    x123=3691.000000
124)    x124=3721.000000
125)    x125=3751.000000
126)    x126=3781.000000
127)    x127=3811.000000
128)    x128=3841.000000
129)    x129=3871.000000
130)    x130=3901.000000
131)    x131=3931.000000
132)    x132=3961.000000
133)    x133=3991.000000
134)    x134=4021.000000
135)    x135=4051.000000
136)    x136=4081.000000
137)    x137=4111.000000
138)    x138=4141.000000
139)    x139=4171.000000
140)    x140=4201.000000
141)    x141=4231.000000
142)    x142=4261.000000
143)    x143=4291.000000
144)    x144=4321.000000
145)    x145=4351.000000
146)    x146=4381.000000
147)    x147=4411.000000
148)    x148=4441.000000
149)    x149=4471.000000
150)    x150=4501.000000
151)    x151=4531.000000
152)    x152=4561.000000
153)    x153=4591.000000
154)    x154=4621.000000
155)    x155=4651.000000
156)    x156=4681.000000
157)    x157=4711.000000
158)    x158=4741.000000
159)    x159=4771.000000
160)    x160=4801.000000
161)    x161=4831.000000
162)    x162=4861.000000
163)    x163=4891.000000
```

```
164)    x164=4921.000000
165)    x165=4951.000000
166)    x166=4981.000000
167)    x167=5011.000000
168)    x168=5041.000000
169)    x169=5071.000000
170)    x170=5101.000000
171)    x171=5131.000000
172)    x172=5161.000000
173)    x173=5191.000000
174)    x174=5221.000000
175)    x175=5251.000000
176)    x176=5281.000000
177)    x177=5311.000000
178)    x178=5341.000000
179)    x179=5371.000000
180)    x180=5401.000000
181)    x181=5431.000000
182)    x182=5461.000000
183)    x183=5491.000000
184)    x184=5521.000000
185)    x185=5551.000000
186)    x186=5581.000000
187)    x187=5611.000000
188)    x188=5641.000000
189)    x189=5671.000000
190)    x190=5701.000000
191)    x191=5731.000000
192)    x192=5761.000000
193)    x193=5791.000000
194)    x194=5821.000000
195)    x195=5851.000000
196)    x196=5881.000000
197)    x197=5911.000000
198)    x198=5941.000000
199)    x199=5971.000000
200)    x200=6001.000000
201)    x201=6031.000000
202)    x202=6061.000000
203)    x203=6091.000000
204)    x204=6121.000000
205)    x205=6151.000000
206)    x206=6181.000000
207)    x207=6211.000000
208)    x208=6241.000000
209)    x209=6271.000000
210)    x210=6301.000000
211)    x211=6331.000000
212)    x212=6361.000000
213)    x213=6391.000000
214)    x214=6421.000000
215)    x215=6451.000000
216)    x216=6481.000000
217)    x217=6511.000000
218)    x218=6541.000000
219)    x219=6571.000000
220)    x220=6601.000000
221)    x221=6631.000000
222)    x222=6661.000000
223)    x223=6691.000000
224)    x224=6721.000000
225)    x225=6751.000000
226)    x226=6781.000000
227)    x227=6811.000000
```

```
228)    x228=6841.000000
229)    x229=6871.000000
230)    x230=6901.000000
231)    x231=6931.000000
232)    x232=6961.000000
233)    x233=6991.000000
234)    x234=7021.000000
235)    x235=7051.000000
236)    x236=7081.000000
237)    x237=7111.000000
238)    x238=7141.000000
239)    x239=7171.000000
240)    x240=7201.000000
241)    x241=7231.000000
242)    x242=7261.000000
243)    x243=7291.000000
244)    x244=7321.000000
245)    x245=7351.000000
246)    x246=7381.000000
247)    x247=7411.000000
248)    x248=7441.000000
249)    x249=7471.000000
250)    x250=7501.000000
251)    x251=7531.000000
252)    x252=7561.000000
253)    x253=7591.000000
254)    x254=7621.000000
255)    x255=7651.000000
256)    x256=7681.000000
257)    x257=7711.000000
258)    x258=7741.000000
259)    x259=7771.000000
260)    x260=7801.000000
261)    x261=7831.000000
262)    x262=7861.000000
263)    x263=7891.000000
264)    x264=7921.000000
265)    x265=7951.000000
266)    x266=7981.000000
267)    x267=8011.000000
268)    x268=8041.000000
269)    x269=8071.000000
270)    x270=8101.000000
271)    x271=8131.000000
272)    x272=8161.000000
273)    x273=8191.000000
274)    x274=8221.000000
275)    x275=8251.000000
276)    x276=8281.000000
277)    x277=8311.000000
278)    x278=8341.000000
279)    x279=8371.000000
280)    x280=8401.000000
281)    x281=8431.000000
282)    x282=8461.000000
283)    x283=8491.000000
284)    x284=8521.000000
285)    x285=8551.000000
286)    x286=8581.000000
287)    x287=8611.000000
288)    x288=8641.000000
289)    x289=8671.000000
290)    x290=8701.000000
291)    x291=8731.000000
```

```
292)    x292=8761.000000
293)    x293=8791.000000
294)    x294=8821.000000
295)    x295=8851.000000
296)    x296=8881.000000
297)    x297=8911.000000
298)    x298=8941.000000
299)    x299=8971.000000
300)    x300=9001.000000
301)    x301=9031.000000
302)    x302=9061.000000
303)    x303=9091.000000
304)    x304=9121.000000
305)    x305=9151.000000
306)    x306=9181.000000
307)    x307=9211.000000
308)    x308=9241.000000
309)    x309=9271.000000
310)    x310=9301.000000
311)    x311=9331.000000
312)    x312=9361.000000
313)    x313=9391.000000
314)    x314=9421.000000
315)    x315=9451.000000
316)    x316=9481.000000
317)    x317=9511.000000
318)    x318=9541.000000
319)    x319=9571.000000
320)    x320=9601.000000
321)    x321=9631.000000
322)    x322=9661.000000
323)    x323=9691.000000
324)    x324=9721.000000
325)    x325=9751.000000
326)    x326=9781.000000
327)    x327=9811.000000
328)    x328=9841.000000
329)    x329=9871.000000
330)    x330=9901.000000
331)    x331=9931.000000
332)    x332=9961.000000
333)    x333=9991.000000
334)    x334=10021.000000
--------------------------------------------------------


Root using False Position Method : --------------------------------------------------------
```

```
N_R(f,10,30,1e-4,x)
```

```
step=1    a=30.000000    f(a)=0.033333
step=2    a=60.000000    f(a)=0.016667
step=3    a=120.000000    f(a)=0.008333
step=4    a=240.000000    f(a)=0.004167
step=5    a=480.000000    f(a)=0.002083
step=6    a=960.000000    f(a)=0.001042
step=7    a=1920.000000    f(a)=0.000521
step=8    a=3840.000000    f(a)=0.000260
step=9    a=7680.000000    f(a)=0.000130
step=10    a=15360.000000    f(a)=0.000065
--------------------------------------------------------

Root using Newton-Raphsonps Method :
```

```
Root is 30720.000000
-------------------------------------------------------
```

```
Secant(f,1,30,1e-4)
```

```
   Xn-1      f(Xn-1)      Xn      f(Xn)      Xn+1      f(Xn+1)
  Column 1

    1.0000

  Column 2

    1.0000

  Column 3

   30.0000

  Column 4

    0.0333

  Column 5

   31.0000

  Column 6

    0.0323

  Column 1

   30.0000

  Column 2

    0.0333

  Column 3

   31.0000

  Column 4

    0.0323

  Column 5

   61.0000

  Column 6

    0.0164

  Column 1

   31.0000

  Column 2

    0.0323

  Column 3
```

```
 61.0000

Column 4

  0.0164

Column 5

 92.0000

Column 6

  0.0109

Column 1

 61.0000

Column 2

  0.0164

Column 3

 92.0000

Column 4

  0.0109

Column 5

153.0000

Column 6

  0.0065

Column 1

 92.0000

Column 2

  0.0109

Column 3

153.0000

Column 4

  0.0065

Column 5

245.0000

Column 6

  0.0041

Column 1
```

```
153.0000
Column 2
  0.0065
Column 3
245.0000
Column 4
  0.0041
Column 5
398.0000
Column 6
  0.0025
Column 1
245.0000
Column 2
  0.0041
Column 3
398.0000
Column 4
  0.0025
Column 5
643.0000
Column 6
  0.0016
 1.0e+03 *
Column 1
  0.3980
Column 2
  0.0000
Column 3
  0.6430
Column 4
  0.0000
```

```
Column 5

  1.0410

Column 6

  0.0000

 1.0e+03 *

Column 1

  0.6430

Column 2

  0.0000

Column 3

  1.0410

Column 4

  0.0000

Column 5

  1.6840

Column 6

  0.0000

 1.0e+03 *

Column 1

  1.0410

Column 2

  0.0000

Column 3

  1.6840

Column 4

  0.0000

Column 5

  2.7250

Column 6

  0.0000

 1.0e+03 *

Column 1
```

```
  1.6840
Column 2
  0.0000
Column 3
  2.7250
Column 4
  0.0000
Column 5
  4.4090
Column 6
  0.0000
 1.0e+03 *
Column 1
  2.7250
Column 2
  0.0000
Column 3
  4.4090
Column 4
  0.0000
Column 5
  7.1340
Column 6
  0.0000
 1.0e+04 *
Column 1
  0.4409
Column 2
  0.0000
Column 3
  0.7134
Column 4
```

```
    0.0000

  Column 5

    1.1543

  Column 6

    0.0000

--------------------------------------------------------

Root using Secant Method :
Root is x = 11543
--------------------------------------------------------
```

Q4.

Find the root of sin(x) function when 0 < x < 2*/pi. Use each routine and comment upon the accuracy of the answer that you find and also the rate of convergence.

```
syms x
f = @(x)sin(x);
Bisection(f,-1,2*pi,1e-4)
```

```
--------------------------------------------------------
Root using Bisection Method :
n = 0
itr = 0
--------------------------------------------------------
```

```
FalsePosition(f,-1,2*pi,1e-4)
```

```
root not located between the entered values
1)    x1=6.283185
--------------------------------------------------------


Root using False Position Method : --------------------------------------------------------
```

```
N_R(f,10,2*pi,1e-4,x)
```

```
--------------------------------------------------------

Root using Newton-Raphsonps Method :
Root is 6.283185
--------------------------------------------------------
```

```
Secant(f,-1,2*pi,1e-4)
```

```
   Xn-1      f(Xn-1)      Xn       f(Xn)      Xn+1      f(Xn+1)
  Column 1

   -1.0000

  Column 2

   -0.8415
```

```
  Column 3

    6.2832

  Column 4

   -0.0000

  Column 5

    6.2832

  Column 6

    0.0000

--------------------------------------------------------

Root using Secant Method :
Root is x = 6.2832
--------------------------------------------------------
```

```
function Bisection(f,xl,xr,tol)
m = (xl + xr) / 2;
n = 0;
error = abs(f(m));
itr = 0;
    if((f(xl)*f(xr)) < 0 )
        while(error >= tol)
            n = (xl + xr) / 2;
            if(f(xl)*f(n)<0)
                xr=n;
            else
                xl=n;
                error = abs(f(n));
            end
            itr = itr + 1;
        end
    end
    fprintf("--------------------------------------------------------\n");
    fprintf("Root using Bisection Method : ");
    n
    itr = itr
    fprintf("--------------------------------------------------------\n\n");
end


function FalsePosition(f,xl,xr,tol)
    i=1;
    itr = 0;
    while(i)
        if f(xl)*f(xr)<0
         i=0;
        else
```

```
        disp('root not located between the entered values');
         i=0;
        end
    end
    if f(xl)<0
     xn=xl;
     xp=xr;
    else
     xn=xr;
     xp=xl;
    end

    xm=xl;
    t=1;
    while (abs(f(xm))>tol)
    xm=(xn*f(xp)-xp*f(xn))/(f(xp)-f(xn));
    fprintf('%d)\tx%d=%f\n',t,t,xm)
    t=t+1;

     if f(xm)<0
      xn=xm;
     else
      xp=xm;
     end
    itr = itr + 1;
    end

    fprintf("--------------------------------------------------------\n");
    fprintf("\n\nRoot using False Position Method : ");
    Root=xm;
    itr = itr;
    fprintf("--------------------------------------------------------\n\n");
end

function N_R(f,N,guess,tol,x)
    step = 1;

    % Finding derivate of given function
    g = diff(f,x);

    % Finding Functional Value
    fa = eval(subs(f,{x},guess));

    while abs(fa)> tol
        fa = eval(subs(f,{x},guess));
        ga = eval(subs(g,{x},guess));
        if ga == 0
            disp('Division by zero.');
            break;
        end

        b = guess - fa/ga;
        fprintf('step=%d\ta=%f\tf(a)=%f\n',step,guess,fa);
        guess = b;
```

```
        if step>N
           disp('Not convergent');
           break;
        end
        step = step + 1;
    end

    fprintf("--------------------------------------------------------\n");
    fprintf("\nRoot using Newton-Raphsonps Method : \n");
    fprintf('Root is %f\n', guess);
    fprintf("--------------------------------------------------------\n\n");
end

function Secant(f,xl,xr,tol)
    c = (xl*f(xr) - xr*f(xl))/(f(xr) - f(xl));
    flag = 0;
    disp('   Xn-1      f(Xn-1)      Xn      f(Xn)      Xn+1      f(Xn+1)');
    disp([xl f(xl) xr f(xr) c f(c)]);
    while (abs(f(c)) > tol)
        xl = xr;
        xr = c;
        c = (xl*f(xr) - xr*f(xl))/(f(xr) - f(xl));
        disp([xl f(xl) xr f(xr) c f(c)]);

        flag = flag + 1;

        if(flag == 100)
            break;
        end
    end

    fprintf("--------------------------------------------------------\n");
    fprintf("\nRoot using Secant Method : \n");
    display(['Root is x = ' num2str(c)]);
    fprintf("--------------------------------------------------------\n\n");
end
```