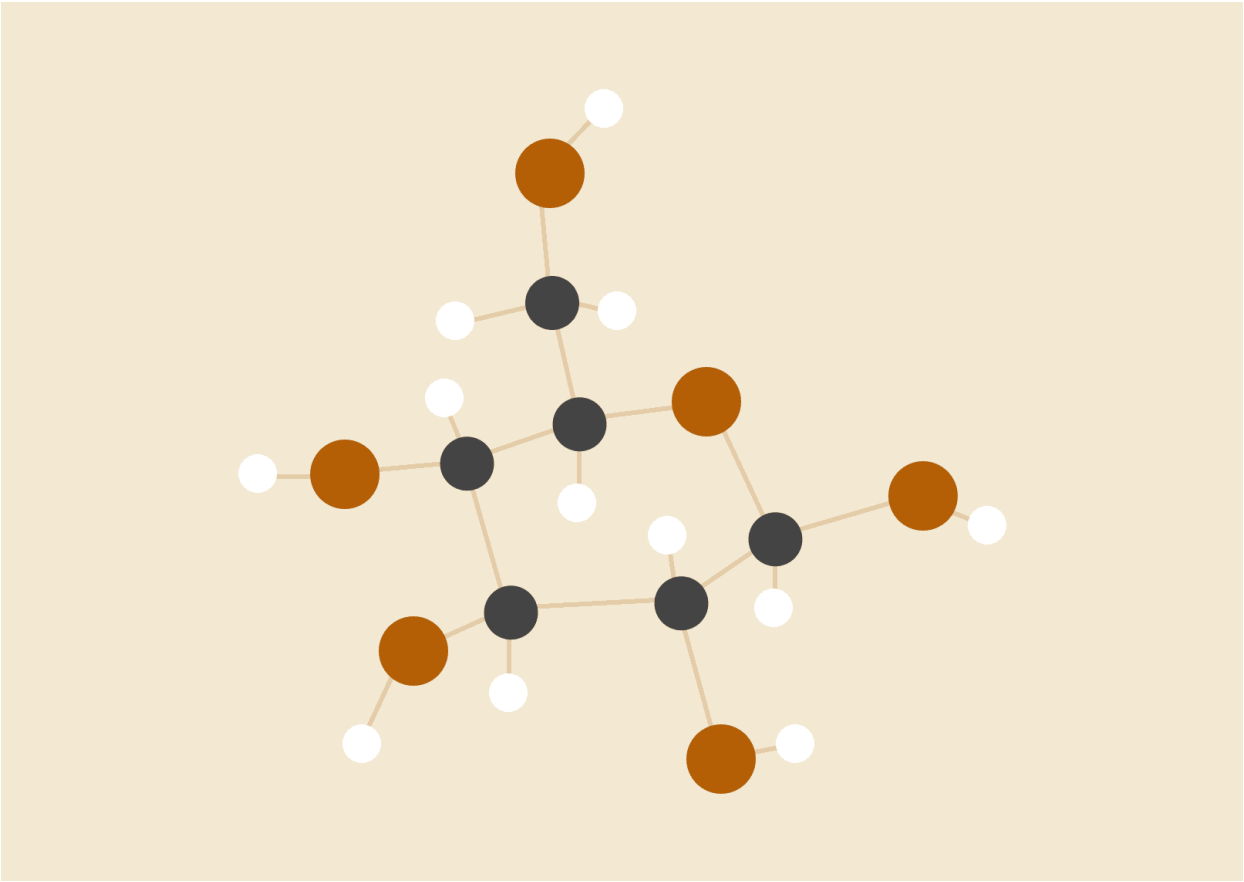
FINAL PROJECT REPORT

MATLAB Programs



Mir Mubashir - K191310
Subhan Saleem - K190325
Jawwad Ali - K191418
Syed Salman Ali - K190151

28.05.2021

CS325 - Numerical Computing

BSCS 4G

RESULTS

Chapter 2:

1. Bisection:

```
Enter an equation eg  "x^(1/2) - cos(x)": x^(1/2) - cos(x)
Enter value of a :0
Enter value of b :1
Enter the tolerance value :0.0001
Solution exists
        n                       a         b                   c                   f(c)                    error

   1.000000000000000                    0   1.000000000000000   0.500000000000000  -0.170475780703825                    0

   2.000000000000000   0.500000000000000   1.000000000000000   0.750000000000000   0.134336534910618   0.304812315614443

   3.000000000000000   0.500000000000000   0.750000000000000   0.625000000000000  -0.020393704463123   0.154730239373741

   4.000000000000000   0.625000000000000   0.750000000000000   0.687500000000000   0.056321251436378   0.076714955899501

   5.000000000000000   0.625000000000000   0.687500000000000   0.656250000000000   0.017806727623804   0.038514523812574

   6.000000000000000   0.625000000000000   0.656250000000000   0.640625000000000  -0.001331824419312   0.019138552043116

   7.000000000000000   0.640625000000000   0.656250000000000   0.648437500000000   0.008227740278852   0.009559564698164

   8.000000000000000   0.640625000000000   0.648437500000000   0.644531250000000   0.003445545257607   0.004782195021245

   9.000000000000000   0.640625000000000   0.644531250000000   0.642578125000000   0.001056259211058   0.002389286046549

  10.000000000000000   0.640625000000000   0.642578125000000   0.641601562500000  -0.000137932657029   0.001194191868088


  11.000000000000000   0.641601562500000   0.642578125000000   0.642089843750000   0.000459125732464   0.000597058389493

  12.000000000000000   0.641601562500000   0.642089843750000   0.641845703125000   0.000160587155484   0.000298538576980

  13.000000000000000   0.641601562500000   0.641845703125000   0.641723632812500   0.000011324904158   0.000149262251326

  14.000000000000000   0.641601562500000   0.641723632812500   0.641662597656250  -0.000063304462642   0.000074629366800
```

2. False Position:

```
Enter an equation eg "-x^3 - cos(x)": -x^3 - cos(x)
Enter the value of a :-1
Enter the value of b :0
Enter the tolerance value :0.000001
Solution exists
        n         a           b              f(a)         f(b)            c           f(c)            err

  Columns 1 through 6

   1.000000000000000  -1.000000000000000                   0   0.459697694131860  -1.000000000000000  -0.685073357326045

  Columns 7 through 8

  -0.452850234475004                   0

  Columns 1 through 6

   2.000000000000000  -1.000000000000000  -0.685073357326045   0.459697694131860  -0.452850234475004  -0.841355125665652

  Columns 7 through 8

  -0.070875968154949   0.381974266320055

  Columns 1 through 6

   3.000000000000000  -1.000000000000000  -0.841355125665652   0.459697694131860  -0.070875968154949  -0.862547487557127


   3.000000000000000  -1.000000000000000  -0.841355125665652   0.459697694131860  -0.070875968154949  -0.862547487557127

  Columns 7 through 8

  -0.008779633363005   0.062096334791944

  Columns 1 through 6

   4.000000000000000  -1.000000000000000  -0.862547487557127   0.459697694131860  -0.008779633363005  -0.865123455684604

  Columns 7 through 8

  -0.001054366765029   0.007725266597976

  Columns 1 through 6

   5.000000000000000  -1.000000000000000  -0.865123455684604   0.459697694131860  -0.001054366765029  -0.865432101825939

  Columns 7 through 8

  -0.000126146724817   0.000928220040212

  Columns 1 through 6

   6.000000000000000  -1.000000000000000  -0.865432101825939   0.459697694131860  -0.000126146724817  -0.865469018788723


  Columns 7 through 8

  -0.000015085680257   0.000111061044560

  Columns 1 through 6

   7.000000000000000  -1.000000000000000  -0.865469018788723   0.459697694131860  -0.000015085680257  -0.865473433482977

  Columns 7 through 8

  -0.000001803974744   0.000013281705513

  Columns 1 through 6

   8.000000000000000  -1.000000000000000  -0.865473433482977   0.459697694131860  -0.000001803974744  -0.865473961398559

  Columns 7 through 8

  -0.000000215721392   0.000001588253352
```

3. Secant Method:

```
Enter equation eg "3*x - exp(x)": 230*x^4 + 18*x^3 + 9*x^2 -221*x - 9
Enter value of a :-1
Enter value of b :0
Enter the tolerance value :0.0000001
Solution exists
              n                pn               err

     0    -1      0

     1     0      0

   2.000000000000000  -0.020361990950226   0.020361990950226

   3.000000000000000  -0.040691256435242   0.020329265485016

   4.000000000000000  -0.040659262577691   0.000031993857551

   5.000000000000000  -0.040659288315725   0.000000025738034

>>
```

Chapter 3:

1. Divided Forward:

```
Enter the values of x and y as matrix: [1.1,1.2,1.3,1.4,1.5; 4,8,12,16,20][1.1,1.2,1.3,1.4,1.5; 4,8,12,16,20]

B =

   4.000000000000000  40.000000000000057  -0.000000000000462   0.000000000003079  -0.000000000015395
   8.000000000000000  39.999999999999964   0.000000000000462  -0.000000000003079                   0
  12.000000000000000  40.000000000000057  -0.000000000000462                   0                   0
  16.000000000000000  39.999999999999964                   0                   0                   0
  20.000000000000000                   0                   0                   0                   0

Enter the value of x to find y :1.2
the interpolated value of x = 1.200000e+00 is 8.0000000 >>
```

2. Divided Backward:

```
Enter the value of x and y as matrix: i.e [1.1,1.2,1.3,1.4,1.5; 4,8,12,16,20][1.1,1.2,1.3,1.4,1.5; 4,8,12,16,20]

B =

   4.000000000000000                   0                   0                   0                   0
   8.000000000000000  40.000000000000057                   0                   0                   0
  12.000000000000000  39.999999999999964  -0.000000000000462                   0                   0
  16.000000000000000  40.000000000000057   0.000000000000462   0.000000000003079                   0
  20.000000000000000  39.999999999999964  -0.000000000000462  -0.000000000003079  -0.000000000015395

Put the value of x to find y: 1.4
The value of y at x = 1.400000e+00 is 16.0000000>>
```

3. Lagrange:

```
Total number of values: 4

X1=  8.1

Y1=  16.94410

X2=  8.3

Y2=  17.56492

X3=  8.6

Y3=  18.50515

X4=  8.7

Y4=  18.82091
Enter the value of x to determine y: 8.4

For the value of 8.40, the value of y is 17.877>>
```

Chapter 4:

Differentiation:

1. Three Point Endpoint (Forward):

```
Enter number of values: 3
Enter Height: 0.1
Enter x(1):0.5
Enter y(1):0.4797
Enter x(2):0.6
Enter y(2):0.5646
Enter x(3):0.7
Enter y(3):0.6442

1.00,    x =0.500,    y=0.4797000

2.00,    x =0.600,    y=0.5646000

3.00,    x =0.700,    y=0.6442000

Enter the number from first col to get answer: 1


Answer using 3 point endpoint (forward method): 0.8755>>
```

2. Three Point Endpoint (Backward):

```
Enter number of values: 3
Enter Height: 0.2
Enter x(1):1
Enter y(1):1
Enter x(2):1.2
Enter y(2):1.2625
Enter x(3):1.4
Enter y(3):1.6595

1.00,    x =1.000,    y=1.0000000

2.00,    x =1.200,    y=1.2625000

3.00,    x =1.400,    y=1.6595000

Enter the number from first col to get answer: 3


Answer using 3 point endpoint (backward method): -2.3213>>
```

3. Three Point Centered:

```
Enter number of values: 3
Enter Height: 0.1
Enter x(): 0.5
Enter y(): 0.4797
Enter x(): 0.6
Enter y(): 0.5646
Enter x(): 0.7
Enter y(): 0.6442
1.00, x =0.500, y=0.4797000
2.00, x =0.600, y=0.5646000
3.00, x =0.700, y=0.6442000
Enter the number from first col to get answer: 2
Answer using 3 point endpoint (forward method): 0.8225000>>
```

Integration:

1. Simpson ⅓:

```
Enter Function eg "@(x)x*log(x)": @(x)x*log(x)
Enter n: 4
Enter a (lower limit): 1
Enter b (upper limit): 2
Area under curve by simpson 1/3rd rule= 0.636310
>>
```

2. Trapezoidal:

```
Enter function: @(x)x^(2)*log(x)
Enter n: 1
Enter Lower Limit: 1
Enter Upper Limit: 1.5
Result using trapezoidal rule is: 0.228074
>>
```

3. Simpson ⅜:

```
Enter Function eg "@(x)x*log(x)": @(x)x*log(x)
Enter n: 4
Enter a (lower limit): 1
Enter b (upper limit): 2
Area under curve by simpson 3/8rd rule= 0.563094
>>
```

Chapter 5:

1. Euler:

```
Enter Function like {@(x,y)0.2*x*y} : @(x,y)0.2*x*y
Enter initial value of x i.e. x0: 1
Enter the final value of x: 1.5
Enter initial value of y i.e. y0: 1
Enter the step length h: 0.1

 x         y
1.0000000  1.0000000
1.1000000  1.0200000
1.2000000  1.0424400
1.3000000  1.0674586
1.4000000  1.0952125 >>
```

2. Heun's:

```
Enter Function like {@(t,y)0.2*t*y} : @(t,y)1+(y/t)
Enter a:  1
Enter b:  2
Enter the initial condition:  2
Enter n: 4
1.0000   2.00000000
1.2500   2.77884615
1.5000   3.60805288
1.7500   4.47913187
2.0000   5.38605331
>>
```

3. Rk4:

```
Enter function eg "@(x,y) (x-y)/2": @(x,y) (x-y)/2
Enter initial value of independent variable : 0
Enter initial value of dependent variable : 1
Enter step size : 0.1
Enter maximum value of independent : 0.2
y(0.10) = 0.9536883
y(0.20) = 0.9145123
>>
```