Name: PARMESH YADAV

Roll No: 2020319

SCIENTIFIC COMPUTING (MTH373)

HOMEWORK – 4

Problem – 1

Problem-1

Given Points $(-2,15), (0,-1), (1,0), (3,-2)$
$\therefore$ 4 Points in total

(a) Monomial Basis

$p_{n-1}(t) = x_1 + x_2 t + x_3 t^2 + x_4 t^3$

The system is:

$$\begin{bmatrix} 1 & -2 & 4 & -8 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 3 & 9 & 27 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 15 \\ -1 \\ 0 \\ -2 \end{bmatrix}$$

using gaussian elimination,

$R_2 \to R_2 - R_1$
$R_3 \to R_3 - R_1$
$R_4 \to R_4 - R_1$

$$\begin{bmatrix} 1 & -2 & 4 & -8 \\ 0 & 2 & -4 & 8 \\ 0 & 3 & -3 & 9 \\ 0 & 5 & 5 & 35 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 15 \\ -16 \\ -15 \\ -17 \end{bmatrix}$$

$R_3 \to R_3 - \frac{3}{2} R_2$

$R_4 \to R_4 - \frac{5}{2} R_2$

$$\begin{bmatrix} 1 & -2 & 4 & 8 \\ 0 & 2 & -4 & 8 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 15 & 15 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 15 \\ -16 \\ 9 \\ 23 \end{bmatrix}$$

$R_4 \to R_4 - 5R_3$

$$\begin{bmatrix} 1 & -2 & 4 & 8 \\ 0 & 2 & -4 & 8 \\ 0 & 0 & 3 & -3 \\ 0 & 0 & 0 & 30 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 15 \\ -16 \\ 9 \\ -22 \end{bmatrix}$$

Now, using back-substitution, $\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} -1 \\ -8/15 \\ 34/15 \\ -11/15 \end{bmatrix}$

$\therefore p_{n-1}(t) = -1 - \frac{8}{15} t + \frac{34}{15} t^2 - \frac{11}{15} t^3$

(b) Lagrange Basis

Directly using the formula,

$$p_{n-1}(x) = \left[\frac{(x-0)(x-1)(x-3)}{(-2-0)(-2-1)(-2-3)} \times 15\right] + \left[\frac{(x+2)(x-1)(x-3)}{(0-(-2))(0-1)(0-3)} \times -1\right]$$

$$+ \left[\frac{(x+2)(x-0)(x-3)}{(1-(-2))(1-0)(1-3)} \times 0\right] + \left[\frac{(x+2)(x-0)(x-1)}{(3-(-2))(3-0)(3-1)} \times (-2)\right]$$

$$p_{n-1}(x) = -1 - \frac{8}{15}x + \frac{34}{15}x^2 - \frac{11}{15}x^3$$

(c) using Newton basis

System is already lower triangular,

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 1 & 3 & 3 & 0 \\ 1 & 5 & 15 & 30 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 15 \\ -1 \\ 0 \\ -2 \end{bmatrix}$$

using forward substitution
$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 15 \\ -8 \\ 3 \\ -11/15 \end{bmatrix}$$

$$\therefore p(x) = x_1 + x_2(x+2) + x_3(x+2)(x) + x_4(x+2)(x-1)(x)$$

$$\therefore p_{n-1}(x) = -1 - \frac{8x}{15} + \frac{34x^2}{15} - \frac{11x^3}{15}$$

(d) We can clearly see that the three approximations give the same polynomial i.e.

$$p_{n-1}(x) = -1 - \frac{8}{15}x + \frac{34}{15}x^2 - \frac{11}{15}x^3$$

Problem – 2

Problem-2

(I) MIDPOINT

(a) $I = \int_0^1 \frac{1}{1+x^2} \cdot dx$

$x_1 = \frac{a+b}{2} = \frac{1+0}{2} = \frac{1}{2}$; $w_1 = b-a = 1-0 = 1$

$\therefore I = f(\frac{1}{2}) \cdot w_1 = \frac{1}{1+(1/2)^2} \cdot (1) = \frac{1}{1+\frac{1}{4}} = \frac{4}{5}$

(b) $I = \int_0^1 \sqrt{x} \cdot \log x \cdot dx$

$x_1 = \frac{a+b}{2} = \frac{1+0}{2} = \frac{1}{2}$; $w_1 = b-a = 1-0 = 1$

$\therefore I = f(\frac{1}{2}) \cdot w_1 = \sqrt{\frac{1}{2}} \cdot \log(\frac{1}{2}) (1)$

$= \frac{-\log(2)}{\sqrt{2}}$

(II) TRAPEZOIDAL

(a) $\int_0^1 \frac{1}{1+x^2} \cdot dx$

$x_1 = 0$; $w_1 = (b-0)/2 = 1/2$
$x_2 = 1$; $w_2 = (b-0)/2 = 1/2$

$I = f(x_1) \cdot w_1 + f(x_2) \cdot w_2$
$= f(0) \cdot \frac{1}{2} + f(1) \cdot \frac{1}{2}$

$= (1)(\frac{1}{2}) + (\frac{1}{2})(\frac{1}{2})$

$= \frac{3}{4}$

(b) $\int_0^1 \sqrt{x} \cdot \log x \cdot dx$

$x_1 = 0$, $w_1 = (b-0)/2 = 1/2$
$x_2 = 1$, $w_2 = (b-0)/2 = 1/2$

$I = f(x_1) \cdot w_1 + f(x_2) \cdot w_2$
$= f(0) \cdot \frac{1}{2} + f(1) \cdot \frac{1}{2}$

$= \sqrt{0} \cdot \log(0) (\frac{1}{2}) + \sqrt{1} \cdot \log(1) \cdot \frac{1}{2}$

$= 0$

(III) SIMPSONS

(a) $\int_0^1 \frac{1}{1+x^2} \cdot dx$

$x_1 = 0$; $w_1 = 1/6$
$x_2 = 1/2$; $w_2 = 4/6$
$x_3 = 1$; $w_3 = 1/6$

$\therefore I = f(x_1) w_1 + f(x_2) w_2 + f(x_3) \cdot w_3$
$= \frac{1}{1+(0)^2} \cdot (\frac{1}{6}) + \frac{1}{1+(\frac{1}{2})^2} \cdot \frac{4}{6} + \frac{1}{1+(1)^2} (\frac{1}{6})$
$= \frac{1}{6} + (\frac{4}{6})(\frac{5}{4}) + \frac{1}{12} = \frac{47}{60}$

(b) $\int_0^1 \sqrt{x} \cdot \log x \cdot dx$

$x_1 = 0$; $w_1 = 1/6$
$x_2 = 1/2$; $w_2 = 4/6$
$x_3 = 1$; $w_3 = 1/6$

$\therefore I = f(x_1) \cdot w_1 + f(x_2) \cdot w_2 + f(x_3) \cdot w_3$
$= \sqrt{0} \cdot \log(0) \cdot (\frac{1}{6}) + \sqrt{\frac{1}{2}} \cdot \log(\frac{1}{2}) (\frac{4}{6}) + \sqrt{1} \cdot \log(1) \cdot \frac{1}{6}$

$= -\frac{\sqrt{2}}{3} \log 2$

(IV) GAUSSIAN QUADRATURE (2-Point)

$\frac{-1}{\sqrt{3}}, \frac{1}{\sqrt{3}}$ in $(-1, 1)$

$I(f) = \frac{b-a}{\beta-\alpha} \sum_{i=1}^{m} w_i \cdot f\left(\frac{(b-a)x_i + a\beta - b\alpha}{\beta-\alpha}\right)$

Acc. to question, $\alpha = -1$, $x_1 = 1/\sqrt{3}$, $w_1 = 1$
$\beta = 1$, $x_2 = -1/\sqrt{3}$, $w_2 = 1$

(a) $\int_0^1 \frac{1}{1+x^2} \cdot dx$

$= \frac{1}{2}\left[f\left(\frac{1+x_1}{2}\right) + f\left(\frac{1+x_2}{2}\right) \right]$

$= \frac{1}{2}\left[f\left(\frac{1+1/\sqrt{3}}{2}\right) + f\left(\frac{1-1/\sqrt{3}}{2}\right) \right]$

$= \frac{1}{2}\left[\frac{1}{1+\left(\frac{1+\sqrt{3}}{2\sqrt{3}}\right)^2} + \frac{1}{1+\left(\frac{1-\sqrt{3}}{2\sqrt{3}}\right)^2} \right]$

$= \frac{1}{2}\left[\frac{12}{12+4+2\sqrt{3}} + \frac{12}{12-4\sqrt{3}+4} \right]$

$= 6\left[\frac{96-12\sqrt{3}+96+12\sqrt{3}}{256-12} \right] = \frac{192}{244} = 0.786$

(b) $$\int_0^1 \sqrt{x} \cdot \log x \cdot dx$$

$$= \frac{1}{2}\left[f\left(\frac{1+\sqrt{3}}{2\sqrt{3}}\right) + f\left(\frac{\cancel{1+\sqrt{3}}\ \sqrt{3}-1}{2\sqrt{3}}\right) \right]$$

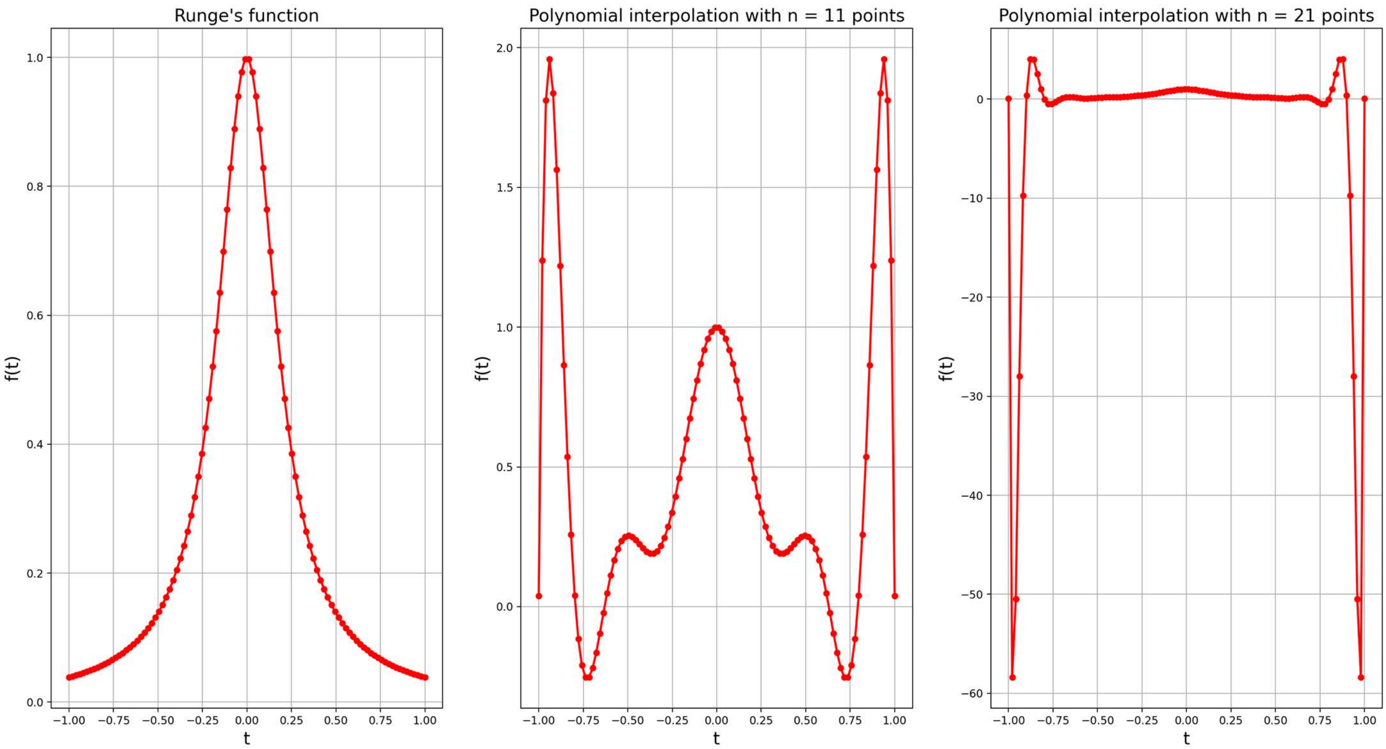
$$= \frac{1}{2}\left[\sqrt{\frac{1+\sqrt{3}}{2\sqrt{3}}} \cdot \log\left(\frac{1+\sqrt{3}}{2\sqrt{3}}\right) + \sqrt{\frac{\sqrt{3}-1}{2\sqrt{3}}} \log\left(\frac{\sqrt{3}-1}{2\sqrt{3}}\right) \right]$$

$$= \frac{1}{2}\left[[0.78]^{1/2} \cdot \log[0.78] + \sqrt{0.21} \log[0.21] \right]$$

$$= -0.4366$$

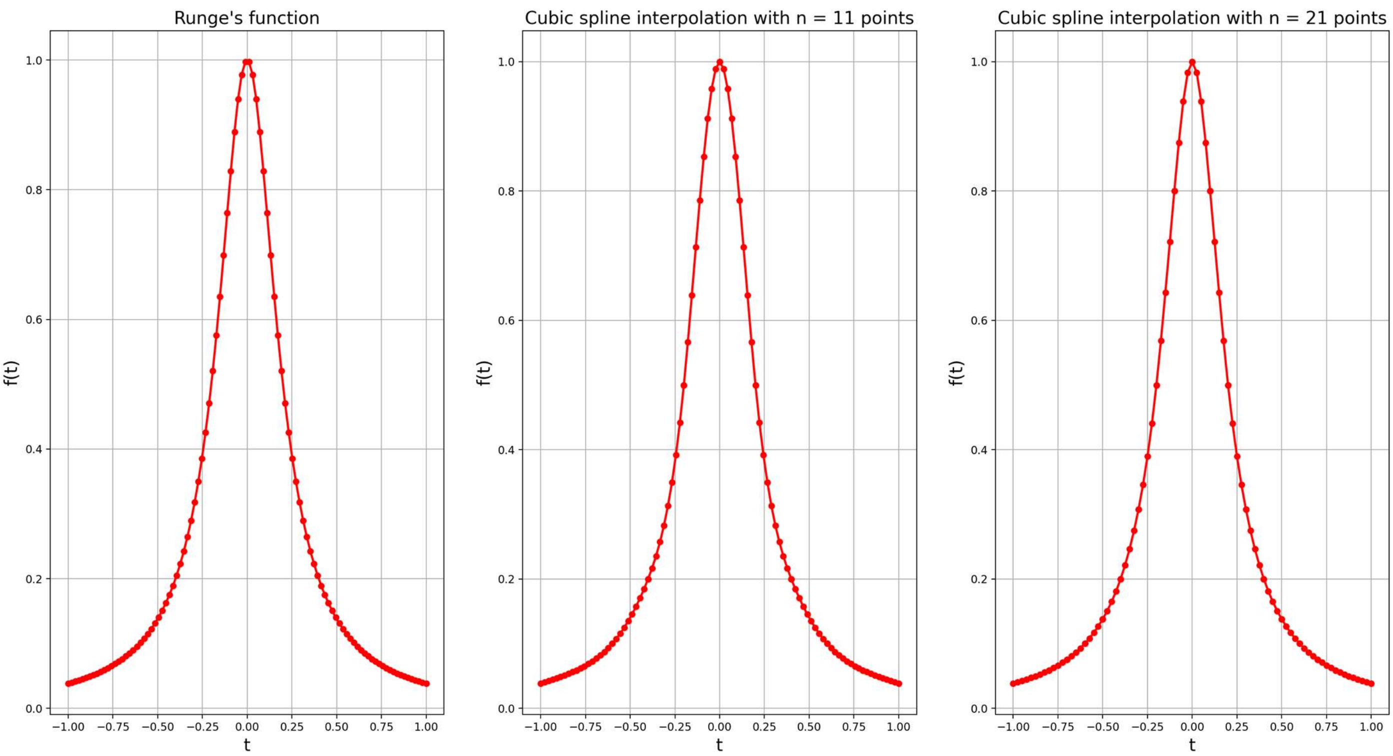
Problem – 3

A. Using polynomial interpolation
Plotting result using 100 points on interval [-1,1]



Using polynomial interpolation, we can observe that output of interpolation matches the original function up to a certain point in the provided domain and becomes less accurate when reaching the endpoints of the domain. Hence, we can say that polynomial interpolation is not very effective at the corners of any given domain. This is further verified by that fact that we studied in class that the vandermond matrix used in polynomial interpolation is ill conditioned at high values of n i.e., in this case n = 21.

B. Using Cubic Spline Interpolation
Plotting result using 100 points on interval [-1,1]



For n = 11, there will be 10 cubic, so 10 points for each polynomial and for n =21, there will be 20 cubic, so 5 points in each polynomial, in total 100 points plotted in each case. As we can see in the above graph, cubic spline interpolation given a way better result than polynomial interpolation mainly because it used piecewise polynomial approach rather than considering the whole domain as a single polynomial.

Problem – 4

Output of calculated values with relative error for each n.

```
[Running] python -u "d:\College\sem 5\Scientific Computing\HomeWork\HomeWork 4\problem_4.py"
n       I                       Relative Error
2       -84.17974381691583      3.478051631368093
4       -29.660256958007253     0.577816182791841
6       -18.714781566369982     0.004427040993211195
8       -17.861876945884624     0.0498140815113577
10      -18.146379303987658     0.03467960626750046
12      -18.40494133986759      0.020925060410242568
14      -18.561234104832184     0.012610862250612549
16      -18.6509684969096       0.0078373238361212
18      -18.703271032796582     0.005055021995635784
20      -18.73476068969875      0.0033798884888308588
22      -18.754408169626863     0.0023347150830324527
24      -18.76709392929351      0.001659879496766987
26      -18.77554528413691      0.001210298616287272
28      -18.781336071704544     0.0009022500937050917
30      -18.78540442668314      0.0006858286267009889
32      -18.788327042134572     0.0005303562731783793
34      -18.790468670250526     0.0004164295629796745
36      -18.79206610109071      0.00033145213901118563
38      -18.79327674009269      0.00026705061303838163
40      -18.794207498805875     0.0002175376852840114
42      -18.794932427672688     0.00017897414555981107
44      -18.795503738545023     0.00014858251607895062
46      -18.79595884881949      0.0001243723294636087
48      -18.796324975583683     0.0001048957371227967
50      -18.796622188551307     8.908510437216773e-05
52      -18.796865474179796     7.614320699702989e-05
54      -18.79706615369522      6.546779756151644e-05
56      -18.797232871758784     5.659901200015543e-05
58      -18.79737229528886      4.9182194865799635e-05
60      -18.7974896137411       4.294128627399191e-05
62      -18.797588901527156     3.765954256503342e-05
64      -18.797673383482692     3.3165414385782936e-05
```

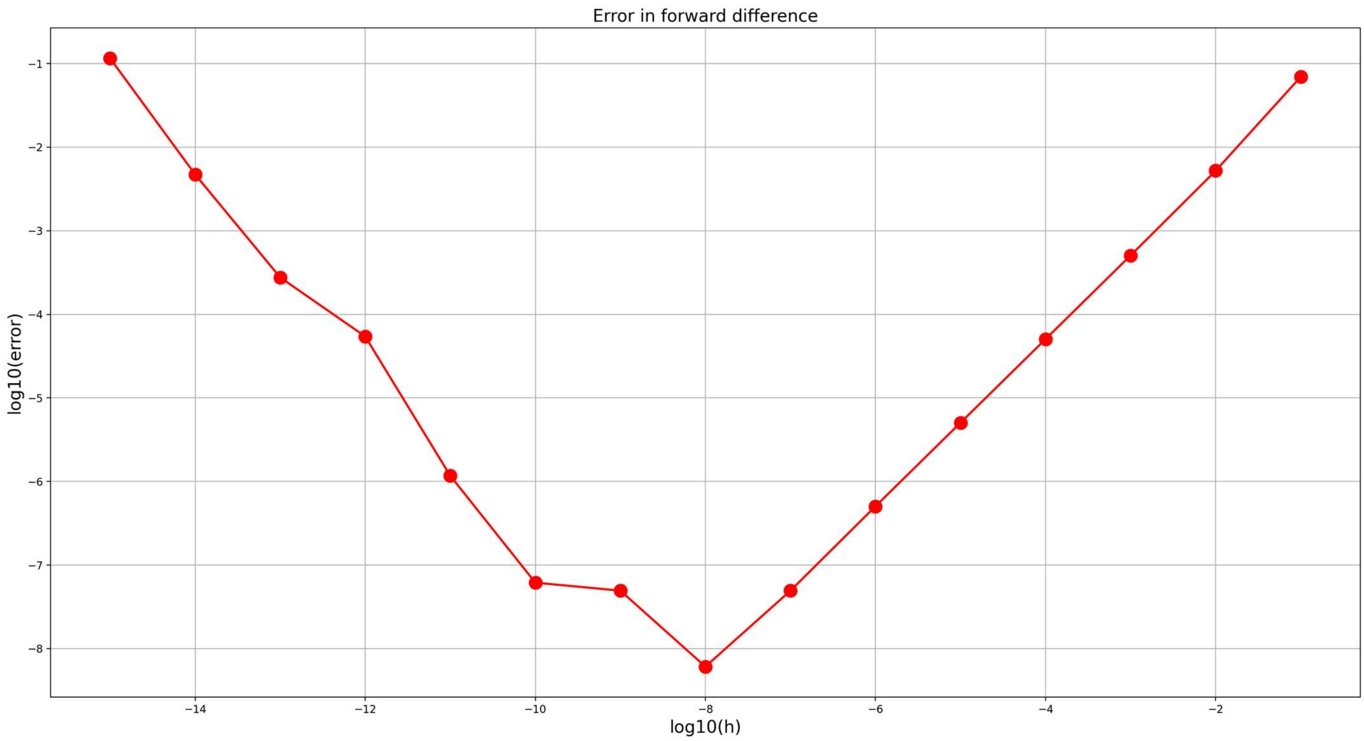
We can clearly see that using composite quadrate we have approximated the function given to us very accurately with very small relative error.

Problem – 5

Using forward difference to find approximation of the derivative of the function given at x = 1, we get the following values using different values of h,

```
[Running] python -u "d:\College\sem 5\Scientific Computing\HomeWork\HomeWork 4\problem_5.py"
Value of H    Calculated Value    Absolute Error
0.1       -0.2589437504200143   0.06940588094341621
0.01        -0.32312262868096076    0.005227002682469728
0.001       -0.3278426595997308   0.0005069717636996818
0.0001        -0.3282990900810301   5.0541282400395904e-05
1e-05       -0.3283445787927164   5.052570714092486e-06
1e-06       -0.32834912611079403    5.052526364512921e-07
1e-07       -0.32834958196836794    4.939506254020287e-08
1e-08       -0.3283496252670659   6.096364579821767e-09
1e-09       -0.32834968077821713    4.941478665143606e-08
1e-10       -0.32834956975591467    6.16075158110796e-08
1e-11       -0.32834845953289005    1.1718305404362361e-06
1e-12       -0.3284039706841213   5.433932069082159e-05
1e-13       -0.32862601528904634    0.0002763839256158529
1e-14       -0.33306690738754696    0.004717276024116479
1e-15       -0.44408920985006256    0.11573957848663208
```

According to the question, here is the log-log plot of this absolute error,



We can clearly see from the above graph that the error in the approximation starts to decrease as the value of h increases but then starts to increase again because of accumulating floating point round off errors cause by rounding of (1+h) in the approximation of the derivative, resulting in the above graph.