NUMERICAL METHODS LAB ASSIGNMENTS

BCSE – II (2ND YEAR, 2ND SEMESTER)

1. Develop a C program to implement **Fixed Point Iteration**. Apply the method on the following equation:

$$e^x - 4x^2 = 0$$

Keep check on whether the condition for convergence is satisfied in your program. Display the output in a tabular form with the following information:

i (Iteration count)	x_i	$\|g(x_i)\|$	$\|f(x_i)\|$	Absolute Error	Order of Convergence

2. Develop a C program to implement **Bisection method**. Test your program to find a root of the following expression.

$$x \sin x + \cos x = 0$$

Precision = 6^{th} place of decimal

Display the output in tabular form with columns:

i (Iteration count)	a (Lower Bound)	b (Upper Bound)	c= (a+b)/2	f(c)	Absolute Error	Order of Convergence

Modify the above program to implement **Regula-Falsi** on the same equation.

Make a comparative assessment of the two methods on the basis of the result you obtain.

3. Develop a C program to implement **Newton-Raphson** method. Test your program on the following equation:

$$e^x = 2x + 1$$

The precision of the solution is 4 places of decimal

Display result in tabular form:

i (Iteration count)	x_i	$f(x_i)$	Absolute Error	Order of Convergence

4. Develop a C program to implement **Secant method**. Test on the same problem as above.

5. The equation $x^3 - 2x^2 - 4x + 8 = 0$ has a double root at x = 2. How would you change your program to apply **Newton-Raphson** in this case?

   The precision of the solution is 3 decimal places

   Take an initial guess of x = 1.2

6. Develop a C program to implement **Gaussian Elimination Method**. Test the program on the following system of linear equations:

$$x_1 + x_2 - x_3 + x_4 = 2$$

$$2x_1 + x_2 + x_3 - 3x_4 = 1$$

$$3x_1 - x_2 - x_3 + x_4 = 2$$

$$5x_1 + x_2 + 3x_3 - 2x_4 = 7$$

   After elimination of each variable, display the augmented co-efficient matrix. Incorporate pivoting in the program.

   (Answer: $x_1 = x_2 = x_3 = x_4 = 1$)

7. Develop a C program to invert a Non-Singular Matrix by **Gauss-Jordan Method**. Arrange for verification of the product of the matrix and the generated inverse.

   Test the program on the following matrix:

$$\begin{matrix} 1 & 5 & 3 \\ 1 & 3 & 2 \\ 2 & 4 & -6 \end{matrix}$$

8. Implement **Gauss-Seidel Method** on the following system of linear equations :

$$5x_1 - x_2 + x_3 = 10$$

$$2x_1 + 8x_2 - x_3 = 11$$

$$-x_1 + x_2 + 4x_3 = 3$$

Display output in tabular form :

i (Iteration count)	x_1	x_2	x_3	Absolute Error

9. Develop a C program to implement **Euler's Method** of numerical solution of first order differential equations.

Test on the following differential equation:

$$y' = 2x^2 + 2y$$

$$y(0) = 1$$

Solution is required over the interval [0, 1] with step length h=0.1

The exact solution is: $y = 1.5e^{2x} - x - x^2 - 0.5$

Produce the output of the program in the following format:

x	y (Computed)	y (Actual Value)	Absolute Error

Modify the above program to implement **Modified Euler's Method.** Produce the output to compare the performance of Euler's method, and modified Euler's method.

10. Develop a C program to implement **Trapezoidal Rule for numerical integration**. In each iteration, the program computes the integral by doubling the number of intervals.

    The program terminates when the desired precision is achieved.

    Test on: $\int_0^1 \frac{1}{1+x} dx$

    The computed value needs to be correct upto the 4^{th} decimal place

11. Develop a C program to implement **Simpson's 1/3rd Rule**. Test on the above problem.