Project Proposal: Scientific Calculator in C

1. Project Title

Scientific Calculator in C

2. Introduction

The Scientific Calculator is a console-based program developed in the C language. It is designed to perform a wide range of mathematical operations, including basic arithmetic (addition, subtraction, multiplication, division) as well as advanced computations such as power, square root, and factorial of a number. This project aims to demonstrate the practical application of fundamental programming concepts in C, including control structures, functions, loops, and mathematical logic.

3. Objectives

• To develop a user-friendly calculator capable of performing basic and scientific calculations.
• To implement modular programming using functions for different operations.
• To provide accurate and efficient results with minimal computational delay.
• To strengthen understanding of C programming concepts such as loops, conditionals, and math functions.

4. Scope of the Project

The calculator will:
• Support basic operations: Addition, Subtraction, Multiplication, Division.
• Perform scientific operations: Power, Square Root, and Factorial.
• Handle integer and floating-point numbers as inputs.
• Provide a simple menu-driven interface for ease of use.

The project is intended for educational purposes, primarily to help beginners understand how to build logic-driven C programs.

5. Methodology

1. Problem Definition: Identify mathematical operations to include in the calculator.
2. Design: Create a structured flow using a menu-driven approach.

3. Implementation: Write modular C functions for each operation.
4. Testing: Verify correctness of outputs for various test cases.
5. Execution: Compile and run using any C compiler such as GCC or Turbo C.

6. Tools and Technologies

• Programming Language: C
• Compiler: GCC / Turbo C / Code::Blocks
• Libraries Used: stdio.h, math.h

7. Features

• Simple and intuitive user interface.
• Performs multiple mathematical functions.
• Uses standard math library for precision.
• Menu-driven program allows continuous calculations without restarting.

8. Expected Outcome

The final output will be a fully functional Scientific Calculator capable of performing both basic and advanced mathematical operations efficiently through a simple console interface.

9. Future Enhancements

• Add trigonometric and logarithmic functions.
• Implement error handling for invalid inputs (like division by zero).
• Develop a GUI-based version using C++ or Python.
• Include memory functions (M+, M-, MR).

10. Conclusion

This project demonstrates the ability to apply C programming fundamentals in a practical context. The Scientific Calculator serves as a foundational tool for students to understand structured programming, modular design, and mathematical logic implementation in C.