Manual String Operations in C

This project demonstrates how to manually implement fundamental string operations in C without using built-in string library functions.   
It is ideal for beginners who want to strengthen their logic-building skills and understand how string manipulation works at a low level.

# 🧠 Features Implemented

- String Length (manual count)  
- String Copy  
- String Compare  
- String Concatenation

All operations are implemented using loops and character arrays — no use of `strlen()`, `strcpy()`, `strcmp()`, or `strcat()`.

# 🛠️ Technologies Used

- Programming Language: C  
- IDE/Compiler: GCC / CodeBlocks / Turbo C / VS Code  
- Platform: Windows / Linux / Mobile C compilers

# 📂 Files in This Repository

- string\_operations.c – Main C source file with all string operations implemented manually  
- output.txt – Sample program output for testing and demonstration  
- sample\_input.txt (optional) – Sample input values used for testing

# 🔍 How It Works

- `fgets()` is used to safely take string input  
- Each string operation is done using a user-defined function  
- Results are displayed with `printf()`

# 🧪 Sample Output

Enter first string: hello  
Enter second string: world  
Concatenated string: helloworld  
Length of String 1: 5  
Length of String 2: 5  
Strings are not same

# 📈 Why This Project is Useful

- Strengthens your understanding of C arrays, loops, and functions  
- Prepares you for programming interviews where logic > libraries  
- Adds a meaningful beginner project to your GitHub portfolio

# 🚀 How to Compile

gcc string\_operations.c -o stringop  
./stringop

# 🏷️ GitHub Tags

#CProgramming #BeginnerProject #StringFunctions #NoLibraryUsed #ManualLogic

# 🙌 Author

Lalitha — Personal Practice Project

This is an individual project created to explore and understand how string operations can be implemented manually using C. It is not affiliated with any college or university.