EEL Experiment no .5

PROBLEM STATEMENT

Develop a C program to implement following operations of strings (without using inbuilt functions) .

1. Copying string.
2. Finding length of the string.
3. Comparison of two strings.
4. Concatenating 2 strings in 1.

RESEARCH

This project focuses on performing basic string operations using the C programming language. String handling is one of the most important aspects of programming, as it allows manipulation and processing of text data. This experiment demonstrates how to manually perform operations like copying, concatenation, length calculation, and comparison without using built-in functions. Understanding these operations strengthens fundamental programming skills and helps students learn how strings are represented and handled in memory.

ANALYSIS

A string in C is a sequence of characters terminated by a null character ('\0'). There are various operations that can be performed on strings. These include: - Copying a string: Duplicates the content of one string into another. - Finding the length: Counts the number of characters before the null terminator. - Concatenation: Joins two strings end-to-end. - Comparison: Checks if two strings are equal or not. These are the foundational operations on which more advanced text manipulation functions are built.

IDEATE

The idea is to create a program that allows the user to input two strings and then performs the following operations: - Copy the first string into another variable. - Find and display the length of the first string. - Concatenate the second string to the first. - Compare both strings to check if they are equal. This helps students understand how strings are processed at the character array level without relying on library functions like strcpy(), strcat(), strlen(), or strcmp().

BUILT

Algorithm:

Step 1: Start

Step 2: Input first string (str1)

Step 3: Input second string (str2)

Step 4: Remove newline characters if present

Step 5: Copy first string into another variable

Step 6: Find and display the length of the first string

Step 7: Concatenate second string to first string

Step 8: Compare both strings and display result

Step 9: End

Test

Case 1:

If both strings are entered correctly without any extra spaces or errors.
Expected Output:
The program will successfully copy the first string, find its length, concatenate both strings, and correctly check if both strings are equal.

Case 2:

If the user enters an empty second string (just presses Enter).
Expected Output:
The program will copy the first string correctly, find its length, but concatenation adds nothing. Comparison will show the strings are not equal unless the first string is also empty.

Case 3:

If both strings entered by the user are exactly the same.
Expected Output:
After comparison, the program will display:
“Strings are equal.”

BUILD:

```
#include <stdio.h>
int main() {
   char str1[100], str2[100], copy[100];
   int i, j, length = 0, flag = 0;

   printf("Enter first string: ");
   fgets(str1, sizeof(str1), stdin);

   printf("Enter second string: ");
   fgets(str2, sizeof(str2), stdin);

   for (i = 0; str1[i] != '\0'; i++) {
      if (str1[i] == '\n')
         str1[i] = '\0';
   }
   for (i = 0; str2[i] != '\0'; i++) {
      if (str2[i] == '\n')
         str2[i] = '\0';
   }
   for (i = 0; str1[i] != '\0'; i++) {
      copy[i] = str1[i];
   }
   copy[i] = '\0';
```

```
printf("\nCopied string: %s", copy);
for (i = 0; str1[i] != '\0'; i++);
length = i;
printf("\nLength of first string: %d", length);
flag = 0;
for (i = 0; str1[i] != '\0' || str2[i] != '\0'; i++) {
   if (str1[i] != str2[i]) {
      flag = 1;
      break;
   }
}

if (flag == 0)
   printf("\nStrings are equal.");
else
   printf("\nStrings are not equal.");
for (i = 0; str1[i] != '\0'; i++);
for (j = 0; str2[j] != '\0'; j++) {
   str1[i + j] = str2[j];
}
str1[i + j] = '\0';

printf("\nAfter concatenation: %s\n", str1);

return 0;
```

}

Output

```
Enter first string: HELLO
Enter second string: WORLD

Copied string: HELLO
Length of first string: 5
Strings are not equal.
After concatenation: HELLOWORLD


=== Code Execution Successful ===
```

Implementation:

https://github.com/P1010-dotcom/String-operations.git