**6b. Write a C program that reads two strings, joins them into another variable, and prints the concatenated result. Use static memory allocation with a fixed size.**

## Description:

This program reads two strings from the user and concatenates them manually without using the built-in strcat() function.

• The program stores both strings in separate character arrays using static memory allocation with a fixed size.

• It then copies the first string into a third array (for the concatenated result).

• After that, it appends the second string character by character to the third array.

• Finally, the program prints the concatenated string.

Example:

Enter first string: Hello

Enter second string: World

Concatenated string: HelloWorld

## Algorithm:

Step 1: Start

Step 2: Declare three character arrays:
   • str1 to store the first string
   • str2 to store the second string
   • result to store the concatenated string (with a fixed size)

Step 3: Prompt the user to enter the first string and store it in str1.

Step 4: Prompt the user to enter the second string and store it in str2.

Step 5: Initialize an index variable i = 0 and copy characters of str1 into result using a loop until '\0' is encountered.

Step 6: Initialize another index variable j = 0 and append characters of str2 into result starting from where str1 ended.

Step 7: After copying both strings, append a null terminator ('\0') at the end of result to mark the end of the string.

Step 8: Print the concatenated string stored in result.

Step 9: Stop.

## Source Code:

```c
#include <stdio.h>

int main() {
    char str1[50], str2[50], result[100];
    int i = 0, j = 0;

    // Taking input for two strings
    printf("Enter first string: ");
    scanf("%s", str1);
    printf("Enter second string: ");
    scanf("%s", str2);

    // Copying first string into result
    while (str1[i] != '\0') {
        result[i] = str1[i];
        i++;
    }
    // Appending second string to result
    while (str2[j] != '\0') {
        result[i] = str2[j];
        i++;
        j++;
    }
    result[i] = '\0'; // Null terminate the concatenated string

    // Displaying the concatenated result
    printf("Concatenated string: %s\n", result);

    return 0;
}
```

## Sample Output:

```
Enter first string: Hello
Enter second string: World
Concatenated String: HelloWorld
```