**F. Y. B. Tech Academic Year 2021-22**

**Subject:** Programming and Problem Solving **Trimester:** 2

**Name:** Shreerang Mhatre  **Division:** 11

**Roll No:** 111056  **Batch:** K3

**ASSIGNMENT NO: 5**

**AIM**: Write a menu driven program to perform all string operations

(user defined function).

**OBJECTIVE:**

To learn string operations in C.

**THEORY:**

**Introduction to Strings**

A string is generally considered as a data type and is often implemented as an array data structure of bytes (or words) that stores a sequence of elements, typically characters, using some character encoding. String may also denote more general arrays or other sequence (or list) data types and structures.

For example, for declaration and initialization create a string consisting of the word "Hello". To hold the null character at the end of the array, the size of the character array containing the string is one more than the number of characters in the word "Hello."

**String Library Functions**

Library functions are inbuilt functions provided by the compilers. These library functions can be easily accessed by importing the particular header file which contains the library function. Similarly, there are many library functions that operate on strings. These library functions are included in the program by importing the header file <string.h>.

There are other string related library functions which operate on the string and produces certain results. Here are some of the commonly used string library functions:

**strlen():**This function returns the length of the string.

**strcat():**This function concatenates two strings.

**strcpy():**This function copies the value of the second string to the first string.

**strcmp():**It compares two strings.

**strlwr():**It changes all the characters of the string to lower case.

**strupr():**It changes all the characters of the string to upper case.

**strchr():**It returns the location or the pointer of the first occurrence of a character in a string.

**strstr():**It returns the location or the pointer of the first occurrence of one string in another.

**User Defined Functions**

A function is a block of code that performs a specific task. C allows you to define functions according to your need. These functions are known as user-defined functions. For example: Suppose, you need to create a circle and color it depending upon the radius and color. You can create two functions to solve this problem:

**createCircle() function**

**color() function**

**Implementation:**

**Platform: 64 –**bit Windows 10**.**

**Technology:** Open Source Visual Studio Code

**Algorithm:**

**Step1) START**

**step2) Declare string 1, string 2, ch, i and result**

**step3) Accept Values in both the strings**

**step4) Make a menu driven program using switch case and assign values of 'ch'**

**step5) If**

**5.1] ch=1, pass a string to the length function and find the length of the string.**

**Print "The length of the string is..."**

**5.2] If ch=2, enter string 1 and string 2 and compare them.**

**5.2.1: If the lengths are equal, use loops to compare characters until the null**

**Character is found.**

**5.2.2: If at any position of the strings the characters are unequal, the loop is**

**Stopped and print " Strings are not same"**

**5.2.3: Else print " Strings are same"**

**5.3] If ch=3, Enter string 1 and string 2. Pass the string to the copy function and**

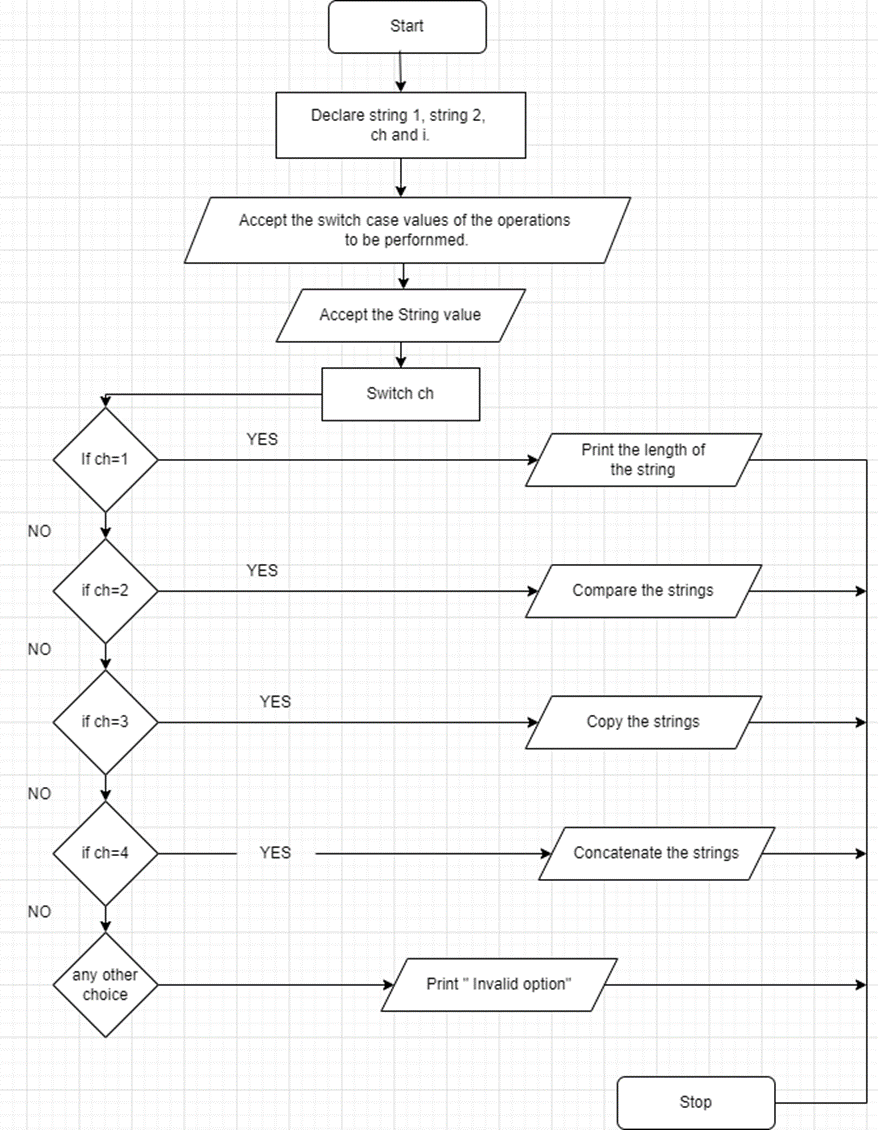
**use Loop until null character is found.**

**5.4] If ch=4, Enter string 1 and string 2. Concatenate the strings and print them.**

**step6) For any other choice, use default to display an " Invalid choice" message.**

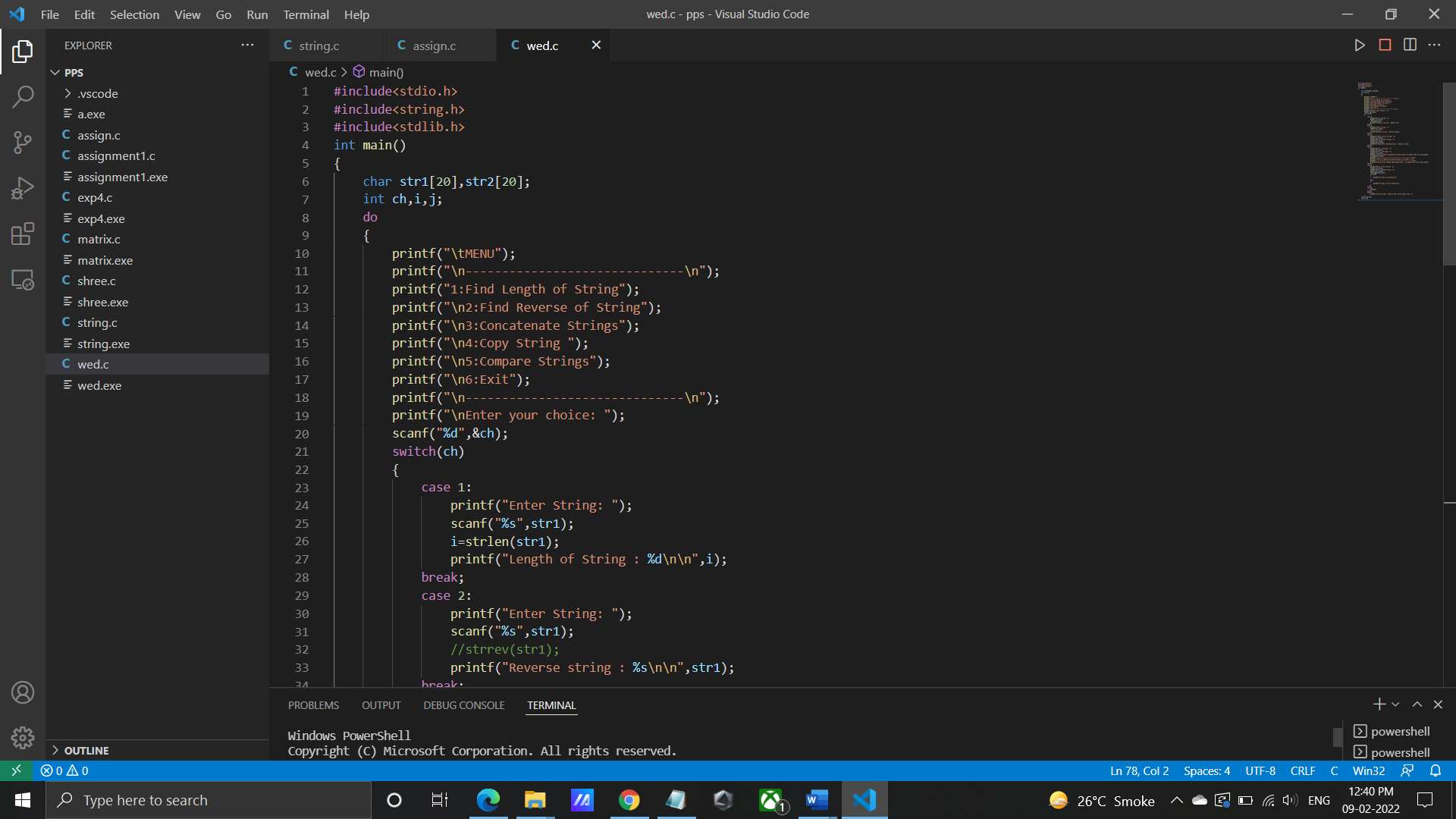
**step7) STOP**

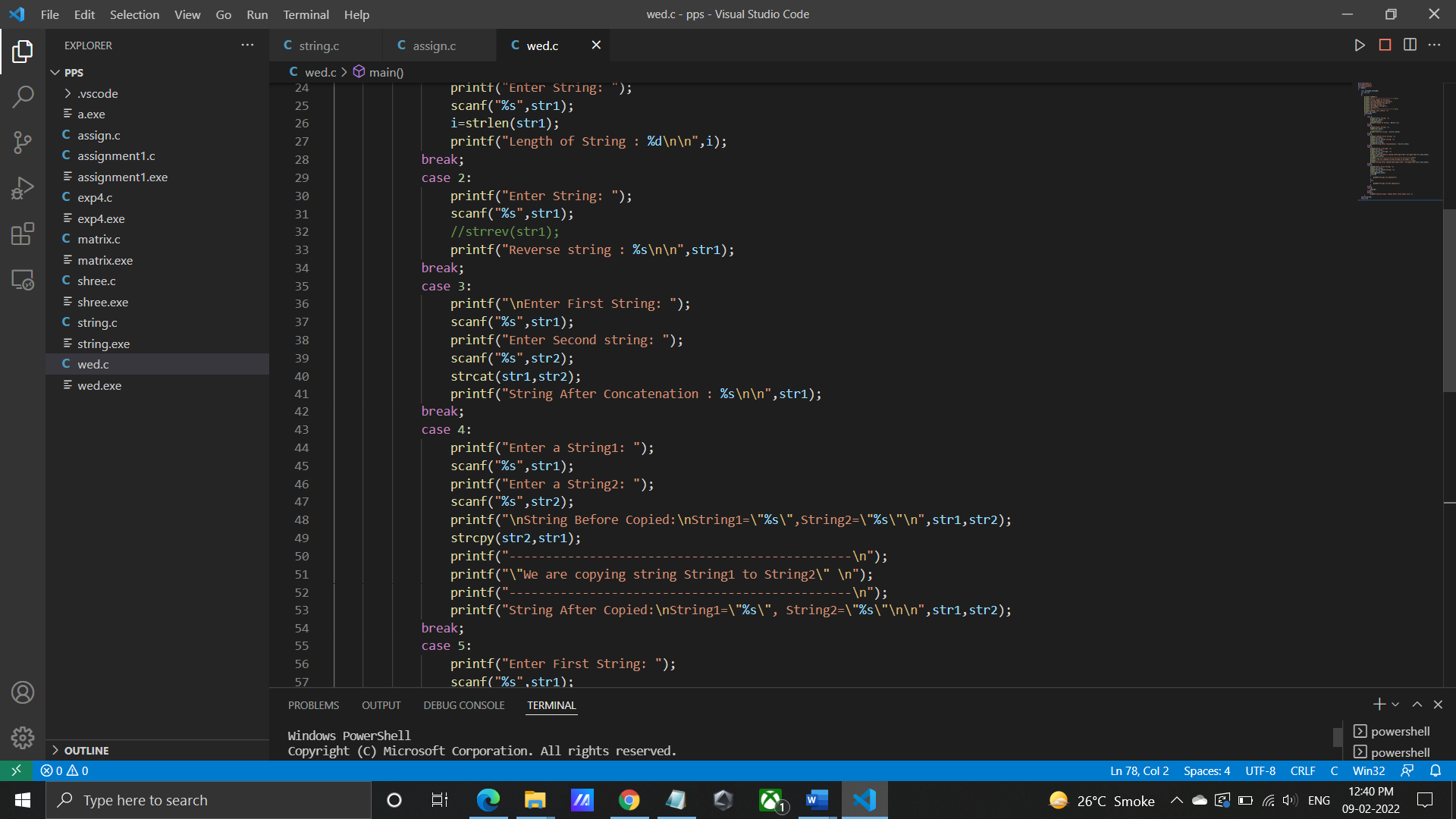
**Flowchart:**

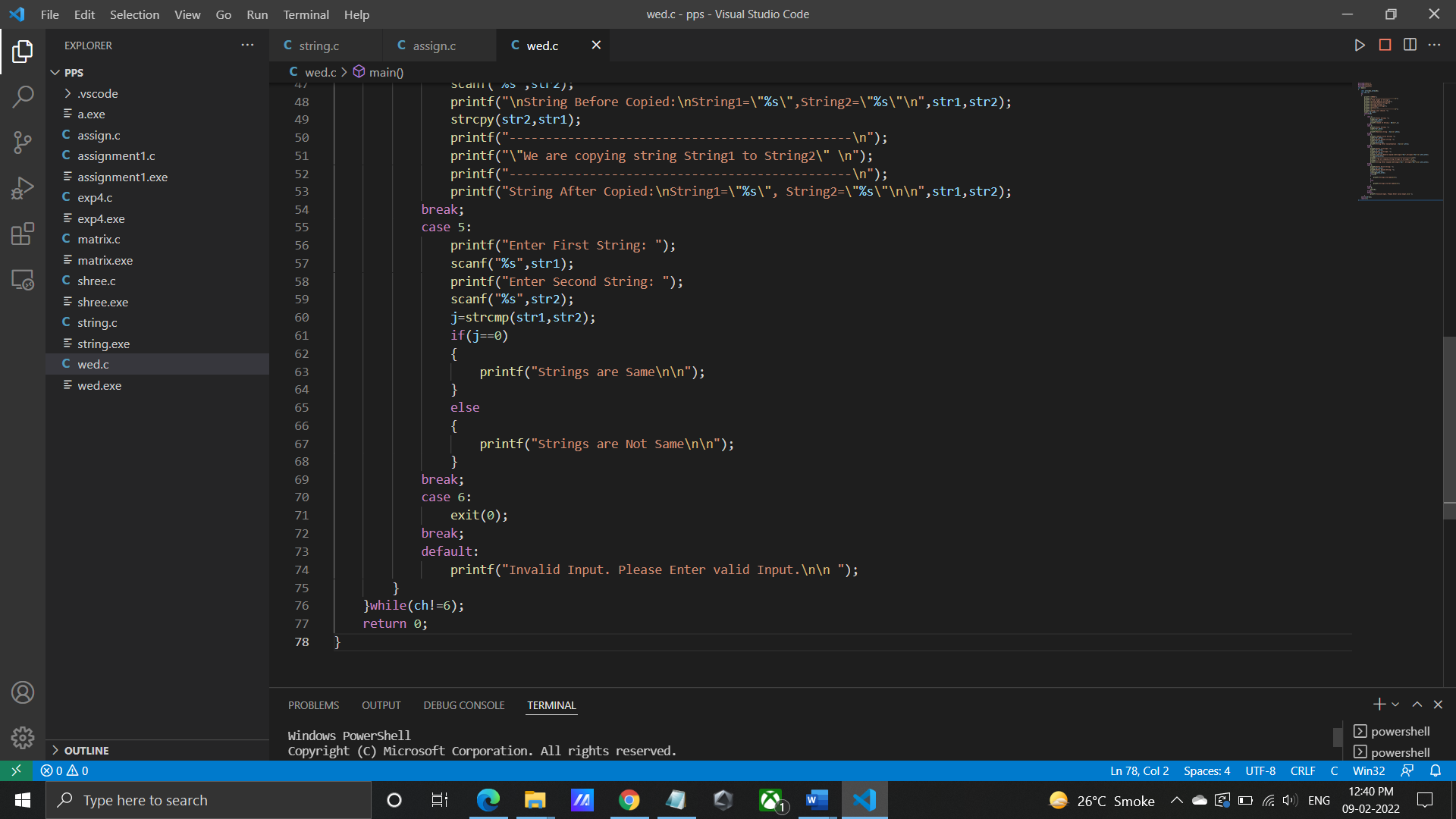
****

**Visual Studio Code :**

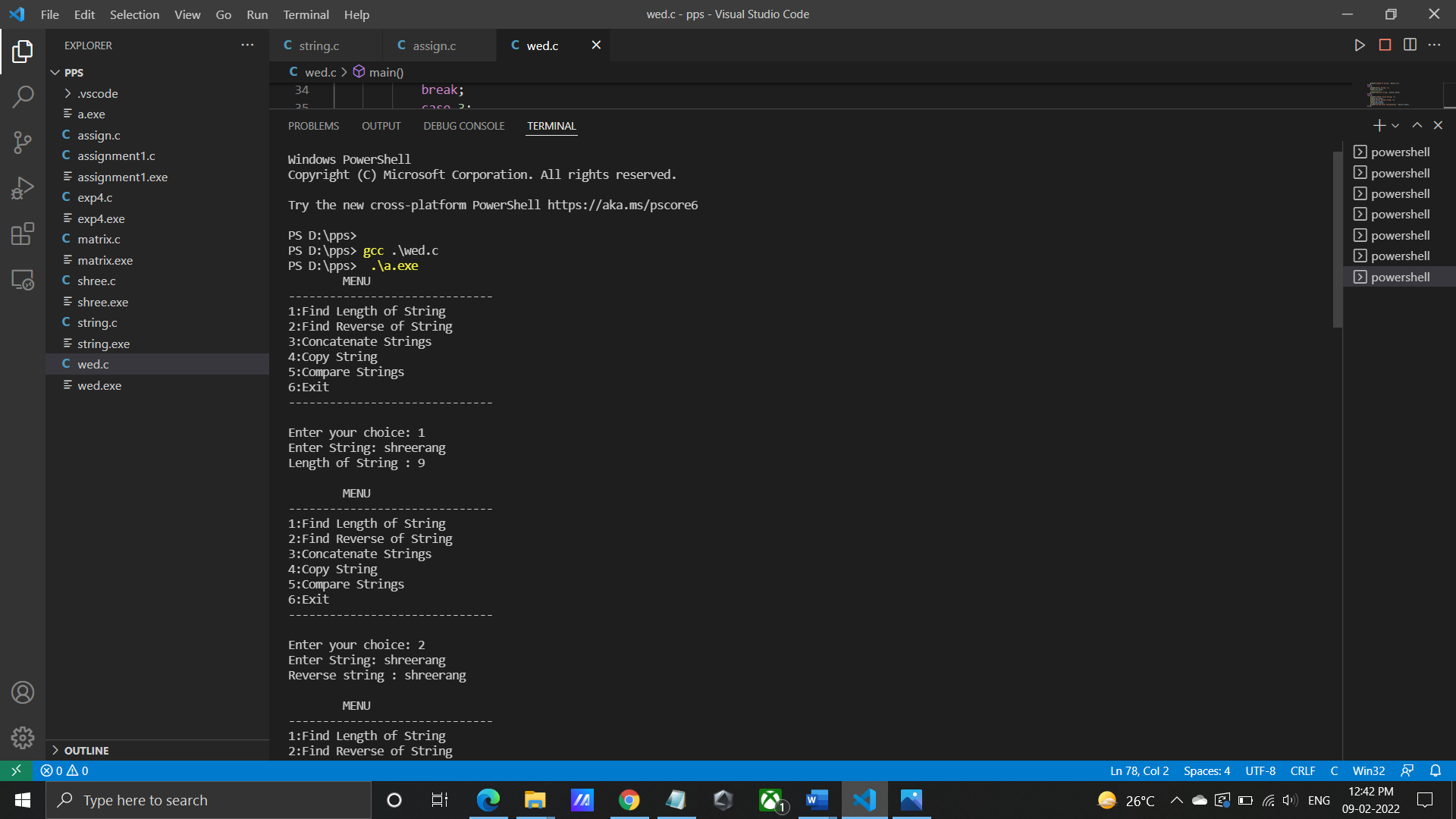
**Code for given program:**

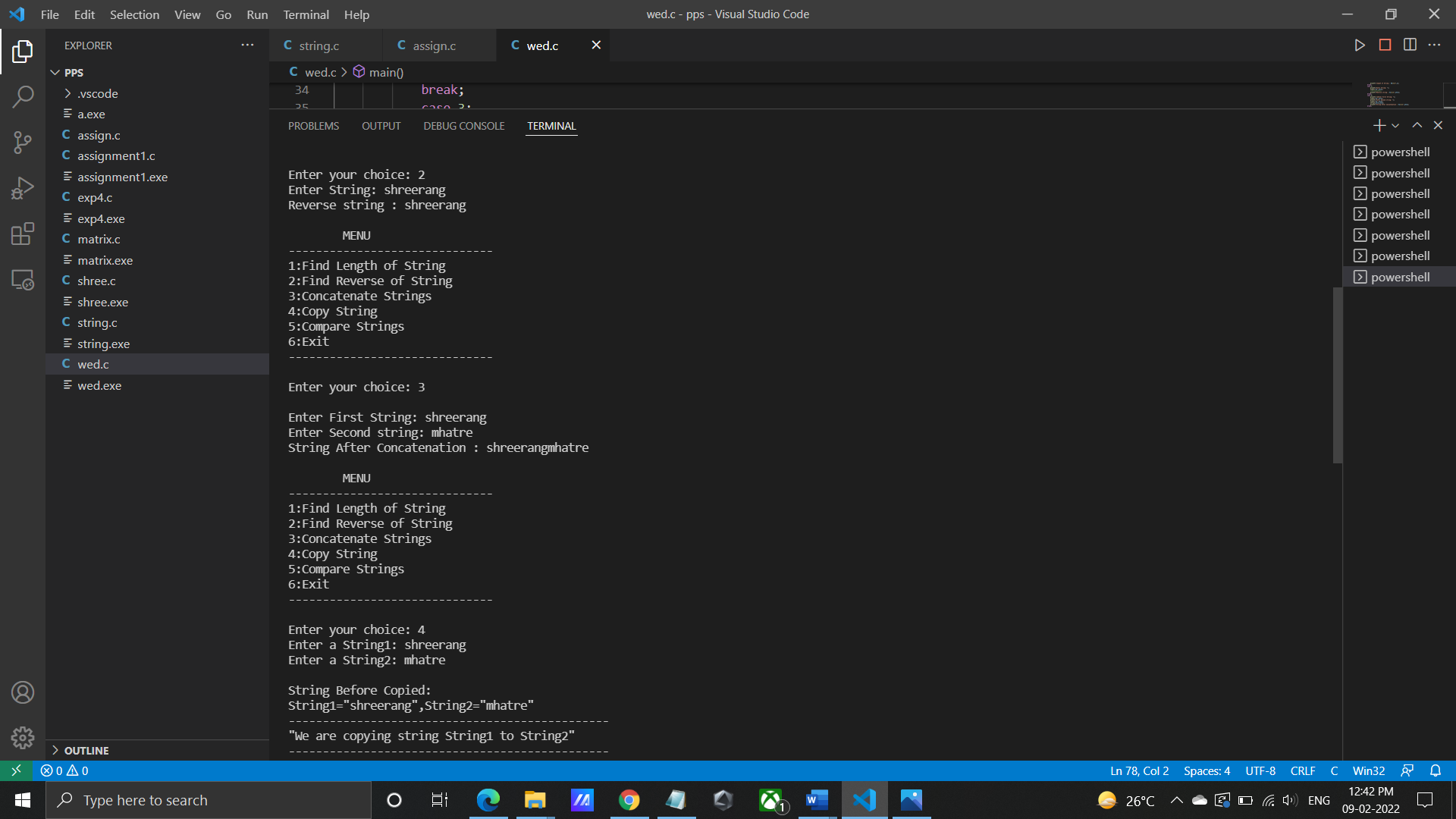
****

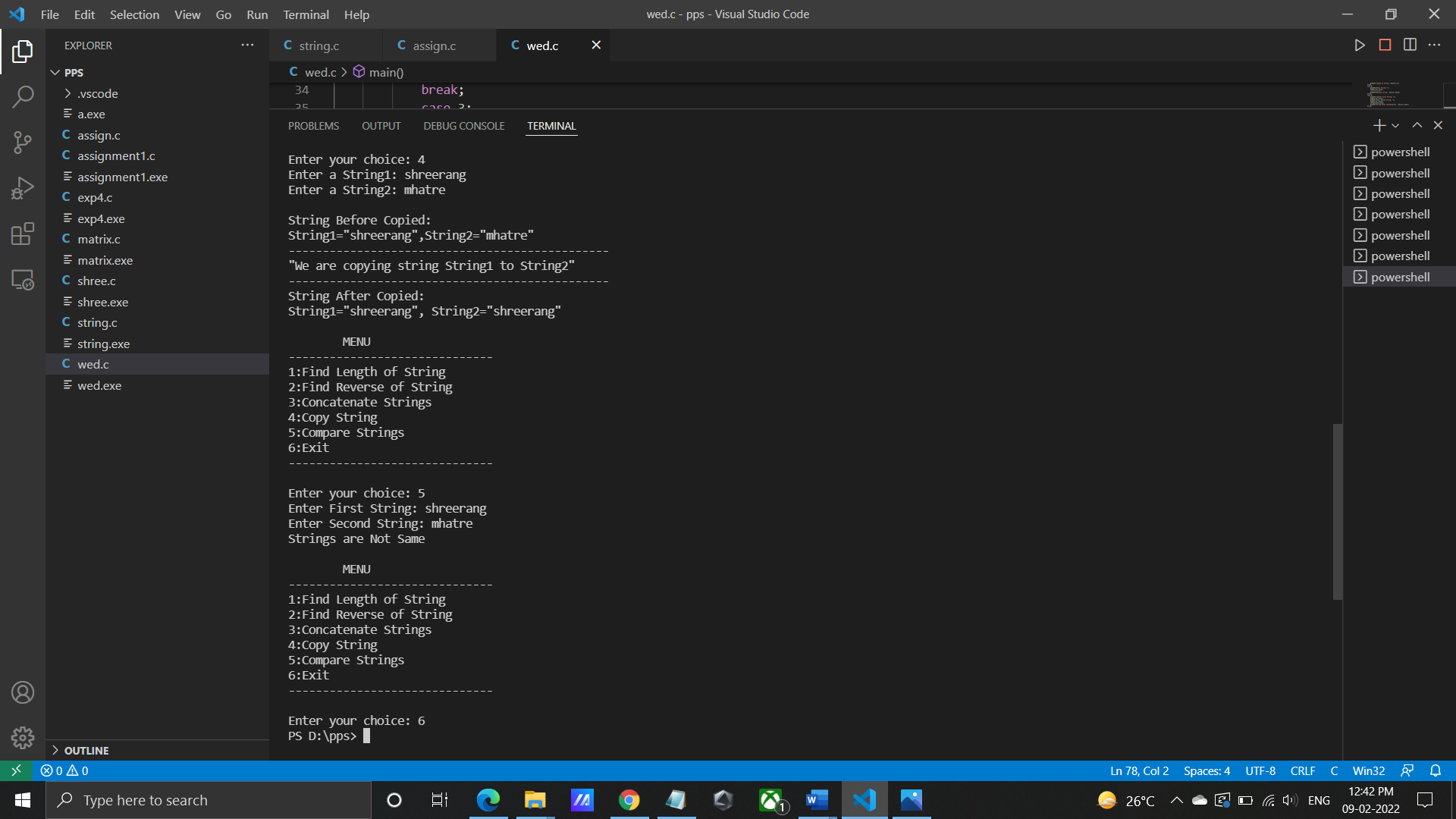
****

****

**Output for given program:**

****

****



**Program/ Code:**

#include<stdio.h>  
#include<string.h>  
#include<stdlib.h>  
int main()  
{  
    char str1[20],str2[20];  
    int ch,i,j;  
    do  
    {  
        printf("\tMENU");  
        printf("\n------------------------------\n");  
        printf("1:Find Length of String");  
        printf("\n2:Find Reverse of String");  
        printf("\n3:Concatenate Strings");  
        printf("\n5:Copy String ");  
        printf("\n5:Compare Strings");  
        printf("\n6:Exit");  
        printf("\n------------------------------\n");  
        printf("\nEnter your choice: ");  
        scanf("%d",&ch);  
        switch(ch)  
        {  
            case 1:  
                printf("Enter String: ");  
                scanf("%s",str1);  
                i=strlen(str1);  
                printf("Length of String : %d\n\n",i);  
            break;  
            case 2:  
                printf("Enter String: ");  
                scanf("%s",str1);  
                //strrev(str1);  
                printf("Reverse string : %s\n\n",str1);  
            break;  
            case 3:  
                printf("\nEnter First String: ");  
                scanf("%s",str1);  
                printf("Enter Second string: ");  
                scanf("%s",str2);  
                strcat(str1,str2);  
                printf("String After Concatenation : %s\n\n",str1);  
            break;  
            case 4:  
                printf("Enter a String1: ");  
                scanf("%s",str1);  
                printf("Enter a String2: ");  
                scanf("%s",str2);

printf("\nString Before Copied:\nString1=\"%s\",String2=\"%s\"\n",str1,str2);

strcpy(str2,str1);  
                printf("-----------------------------------------------\n");  
                printf("\"We are copying string String1 to String2\" \n");  
                printf("-----------------------------------------------\n");  
                printf("String After Copied:\nString1=\"%s\", String2=\"%s\"\n\n",str1,str2);  
            break;  
            case 5:  
                printf("Enter First String: ");  
                scanf("%s",str1);  
                printf("Enter Second String: ");  
                scanf("%s",str2);  
                j=strcmp(str1,str2);  
                if(j==0)  
                {  
                    printf("Strings are Same\n\n");  
                }  
                else  
                {  
                    printf("Strings are Not Same\n\n");  
                }  
            break;  
            case 6:  
                exit(0);  
            break;  
            default:  
                printf("Invalid Input. Please Enter valid Input.\n\n ");  
        }  
    }while(ch!=6);  
    return 0;  
}

**INPUT:** String

**OUTPUT :**

**Windows PowerShell**

**Copyright (C) Microsoft Corporation. All rights reserved.**

**Try the new cross-platform PowerShell https://aka.ms/pscore6**

**PS D:\pps>**

**PS D:\pps> gcc .\wed.c**

**PS D:\pps> .\a.exe**

**MENU**

**------------------------------**

**1:Find Length of String**

**2:Find Reverse of String**

**3:Concatenate Strings**

**4:Copy String**

**5:Compare Strings**

**6:Exit**

**------------------------------**

**Enter your choice: 1**

**Enter String: shreerang**

**Length of String : 9**

**MENU**

**------------------------------**

**1:Find Length of String**

**2:Find Reverse of String**

**3:Concatenate Strings**

**4:Copy String**

**5:Compare Strings**

**6:Exit**

**------------------------------**

**Enter your choice: 2**

**Enter String: shreerang**

**Reverse string : shreerang**

**MENU**

**------------------------------**

**1:Find Length of String**

**2:Find Reverse of String**

**3:Concatenate Strings**

**4:Copy String**

**5:Compare Strings**

**6:Exit**

**------------------------------**

**Enter your choice: 3**

**Enter First String: shreerang**

**Enter Second string: mhatre**

**String After Concatenation : shreerangmhatre**

**MENU**

**------------------------------**

**1:Find Length of String**

**2:Find Reverse of String**

**3:Concatenate Strings**

**4:Copy String**

**5:Compare Strings**

**6:Exit**

**------------------------------**

**Enter your choice: 4**

**Enter a String1: shreerang**

**Enter a String2: mhatre**

**String Before Copied:**

**String1="shreerang",String2="mhatre"**

**-----------------------------------------------**

**"We are copying string String1 to String2"**

**-----------------------------------------------**

**String After Copied:**

**String1="shreerang", String2="shreerang"**

**MENU**

**------------------------------**

**1:Find Length of String**

**2:Find Reverse of String**

**3:Concatenate Strings**

**4:Copy String**

**5:Compare Strings**

**6:Exit**

**------------------------------**

**Enter your choice: 5**

**Enter First String: shreerang**

**Enter Second String: mhatre**

**Strings are Not Same**

**MENU**

**------------------------------**

**1:Find Length of String**

**2:Find Reverse of String**

**3:Concatenate Strings**

**4:Copy String**

**5:Compare Strings**

**6:Exit**

**------------------------------**

**Enter your choice: 6**

**PS D:\pps**

**CONCLUSION:** Thus we have learned various string operations in C.

**FAQ:**

1. How to find the maximum occurring character in given String?

ANS: If there are more than one characters occuring the maximum number of time then

any of the character is returned. In this post, the lexicographically smallest character

of all characters is returned.

1. How to remove all duplicates from a given string?

ANS: Sort the elements. Now in a loop, remove duplicates by comparing the current

character with previous character.

1. How do you check if a given String is Palindrome or not?

ANS: A string is said to be palindrome if it reads the same backward as forward. For e.g.

above string is a palindrome because if we try to read it from backward, it is same

as forward. One of the approach to check this is iterate through the string till middle

of string and compare a character from back and forth.