

Computer Engineering Department, S.V.N.I.T.
Surat. B Tech (CO) –**II Year semester-III**

Course:
Data Structures
Assignment-II

Note: Write all the program logic/functions in the user-defined header file “mystring.h”.

Your user-defined function should work similarly to the in-built function of the library header file string.h.

1. Write a program to find the length of a given string.
 2. Write a program to concatenate two given string.
 3. Write a program to copy one string to another string.
 4. Write a program to compare two given string.
 5. Write a program to search for the first occurrence of the character ‘c’ in the given string.
 6. Write a program to find sub string is there in given string or not?
 7. Write a program to generate reverse of a string.
-

A.) Header File: “mystring.h”

```
//Function 1 : Write a program to find the length of a given string.

int mystrlen(char str[1000000])
{
    int len = 0;
    for (; str[len] != '\0'; ++len)
    {
        ;
    }
    return len;
}

//Function 2 : Write a program to concate two given string.

static char *mystrconcat(char *str1, char *str2)
{
    int i = 0, j = 0;
    while (str1[i++] != '\0')
        ;
}
```

```

    i--;
    while (str2[j] != '\0')
    {
        str1[i++] = str2[j++];
    }
    str1[i] = '\0';

    return str1;
}

```

//Function 3 : Write a program to copy one string to another string.

```

void mystrcopy(char *str1, char *str2)
{
    int i = 0;
    while (str1[i] != '\0')
    {
        str2[i] = str1[i];
        i++;
    }
    return;
}

```

//Function 4 : Write a program to compare two given string.

```

int mystrcomp(char *str1, char *str2)
{
    int i = 0, j = 0;
    while (str1[i] != '\0' && str2[i] != '\0')
    {
        if (str1[i] == str2[i])
        {
            i++;
            continue;
        }
        else
        {
            return (int)(str1[i] - str2[i]);
        }
    }
    return 0;
}

```

*//Function 5 : Write a program to search for the first occurrence
// of the character 'c' (any char) in the given string.*

```

int mystroccur(char *str1, char ch)
{
    int i = 0;
    while (str1[i] != '\0')

```

```

{
    if (str1[i] == ch)
    {
        return i;
    }
    i++;
}
return -1;
}

//Function 6 : Write a program to find sub string is there in given string or not?

```

```

int mystrsub(char *str1, char *sub)
{
    int i = 0, j = 0;
    while (str1[i] != '\0')
    {
        if (str1[i] == sub[j])
        {
            int k = j;
            int oldi = i;
            int flag = 1;
            while (str1[i] != '\0' && sub[k] != '\0')
            {
                if (str1[i] == sub[k])
                {
                    k++;
                    i++;
                }
                else
                {
                    flag = 0;
                    break;
                }
            }
            if (flag)
            {
                return 1; // found
            }
            i = oldi;
        }

        i++;
    }

    return 0; // substring not found
}

```

```

//Function 7 : Write a program to generate reverse of a string.

```

```

char *mystrev(char *str1, char *str2)
{
    int len = 0;
    for (; str1[len] != '\0'; ++len)
    {
        ;
    }
    int i = 0;
    for (int j = len - 1; j >= 0; j--)
    {
        str2[i] = str1[j];
        i++;
    }
    str2[i] = '\0';
    return str2;
}

```

B.) C Program File (“A.c”)

```

#include <stdio.h>
#include "mystring.h"

#define MAX 100000

int main()
{
    //Function 1 : Write a program to find the length of a given string.

    printf("\n1 - MYSTRLEN FUNCTION : \n\n");

    char string[MAX];

    printf("Enter String :\n");
    scanf("%s", string);

    printf("Length of String : %d\n", mystrlen(string));

    //Function 2 : Write a program to concat two given string.

    printf("\n2 - MYSTRCONCAT FUNCTION : \n\n");

    char string1[MAX];
    char string2[MAX];

    printf("Enter String 1 :\n");
    scanf("%s", string1);

    printf("Enter String 2 :\n");

```

```

scanf("%s", string2);

// char string3[MAX];
char *str3 = mystrconcat(string1, string2);

printf("Concatanated String : %s\n", str3);

//Function 3 : Write a program to copy one string to another string.

printf("\n3 - MYSTRCOPY FUNCTION : \n\n");

char string3[MAX];
char string4[MAX];

printf("Enter String to be Copied :\n");
scanf("%s", string3);

mystrcopy(string3, string4);

printf("Copied String : %s\n", string4);

//Function 4 : Write a program to compare two given string.

printf("\n4 - MYSTRCOMP FUNCTION : \n\n");

char string5[MAX];
char string6[MAX];

printf("Enter String 5 :\n");
scanf("%s", string5);

printf("Enter String 6 :\n");
scanf("%s", string6);

int cmp = mystrcomp(string5, string6);

if (cmp == 0)
{
    printf("Result : Both Strings are Equal");
}
else
{
    if (cmp > 0)
    {
        printf("Result : %s is greater than %s\n", string5, string6);
    }
    else
    {
        printf("Result : %s is less than %s\n", string5, string6);
    }
}

```

```

}

//Function 5 : Write a program to search for the first
//occurrence of the character 'c' (any char) in the given string.

printf("\n5 - MYSTROCCUR FUNCTION : \n\n");

char string7[MAX], ch;

printf("Enter String 7 :\n");
scanf("%s", string7);

fflush(stdin); // So that it does not count "Enter" as Character

printf("Enter Character to Search :\n");
ch = getchar();

int srch = mystroccur(string7, ch);

if (srch == -1)
{
    printf("Result : %c not found in %s\n", ch, string7);
}
else
{
    printf("Result : %c found in %s at index %d\n", ch, string7, srch);
}

//Function 6 : Write a program to find sub string is there in given string or not?

printf("\n6 - MYSTRSUB FUNCTION : \n\n");

char parent[MAX];
char substring[MAX];

printf("Enter Parent String :\n");
scanf("%s", parent);
printf("Enter Sub-String to be Found :\n");
scanf("%s", substring);

if (mystrsub(parent, substring))
{
    printf("Result : %s found as substring in %s\n", substring, parent);
}
else
{
    printf("Result : %s not found as substring in %s\n", substring, parent);
}

//Function 7 : Write a program to generate reverse of a string.

```

```

printf("\n7 - MYSTRREV FUNCTION : \n\n");

char string8[MAX];
char string9[MAX];

printf("Enter String 8 :\n");
scanf("%s", string8);

mystrrev(string8, string9);

printf("Reversed String : %s\n", string9);

return 0;
}

```

C.) Sample Test Cases

1.) [DataStructure], [Hello_ , World!]

```

1 - MYSTRLEN FUNCTION :

Enter String :
DataStructure
Length of String : 13

2 - MYSTRCONCAT FUNCTION :

Enter String 1 :
Hello_
Enter String 2 :
World!
Concatanated String : Hello_World!

```

2.) [Computer_Science], [Apple, Apple]

```

3 - MYSTRCOPY FUNCTION :

Enter String to be Copied :
Computer_Science
Copied String : Computer_Science

4 - MYSTRCOMP FUNCTION :

Enter String 5 :
Apple
Enter String 6 :
Apple
Result : Both Strings are Equal

```

3.) [abcdef, abdced]

```
4 - MYSTRCOMP FUNCTION :  
  
Enter String 5 :  
abcdef  
Enter String 6 :  
abdcef  
Result : abcdef is less than abdcef
```

4.) [DSA, CS], [computer_science, z]

```
4 - MYSTRCOMP FUNCTION :  
  
Enter String 5 :  
DSA  
Enter String 6 :  
CS  
Result : DSA is greater than CS  
  
5 - MYSTROCCUR FUNCTION :  
  
Enter String 7 :  
computer_science  
Enter Character to Search :  
z  
Result : z not found in computer_science
```

5.) [Elephant, h], [Bat_and_Ball, and]

```
5 - MYSTROCCUR FUNCTION :  
  
Enter String 7 :  
Elephant  
Enter Character to Search :  
h  
Result : h found in Elephant at index 4  
  
6 - MYSTRSUB FUNCTION :  
  
Enter Parent String :  
Bat_and_Ball  
Enter Sub-String to be Found :  
and  
Result : and found as substring in Bat_and_Ball
```


6.) [Nobita_and_Doraemon, Tom]

```
6 - MYSTRSUB FUNCTION :
```

```
Enter Parent String :
```

```
Nobita_and_Doraemon
```

```
Enter Sub-String to be Found :
```

```
Tom
```

```
Result : Tom not found as substring in Nobita_and_Doraemon
```

7.) [ecnalubmA]

```
7 - MYSTRREV FUNCTION :
```

```
Enter String 8 :
```

```
ecnalubmA
```

```
Reversed String : Ambulance
```

Submitted By:

Roll Number: **U19CS012** (D-12)

Name: *Bhagya Rana*