# Assignment 3-String Operations

Write C++ program for string operations- copy, concatenate, check substring, equal, reverse and length

## Concepts Used

* ”string” as a datatype
* Manipulation of string

## Theory of Concepts Used

### ”string” as a datatype

The C-style character string originated within the C language and continues to be supported within C++. This string is actually a one-dimensional array of characters which is terminated by a null character ’—0’. Thus a null-terminated string contains the characters that comprise the string followed by a null.

The following 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.”

char

greeting[6] = {

’H’

,

’e’

,

’l’

,

’l’

,

’o’

,

’\0’

}

;

If you follow the rule of array initialization, then you can write the above statement as follows:

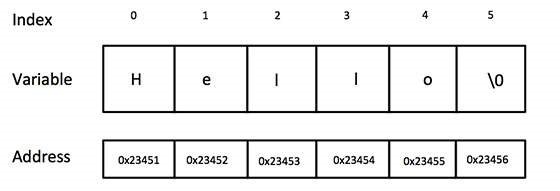
char

greeting[] =

"Hello"

;

Following is the memory presentation of above defined string in C/C++:



### Manipualtion of a string

Manipulating of strings in C++ by operator overloading using character arrays, pointers and string functions. There are no operators for manipulating the strings. There are no direct operator that could act upon the strings or manipulate the strings. Although there are these limitations exist in C++, it permits us to create our own definitions of operators that can be used to manipulate the strings very much similar to the decimal number. We can manipulate strings by operator overloading as this is not achieved by operators only.

String Manipulation in C++ contains many string functions which we can use to manipulate the strings. In some compilers we can use them only by including iostream but in compilers which gives error have to include string library. Those functions can be Stringname.length , Stringname[expression](this expression is a number of character to show from the string), and many more.

## Algorithm

get\_length algorithm-

1.Start.

2.Declare a varible length.

3.Initialize it to zero.

4.Traverse through each character of string and increment length by one.

5.Continue to traverse till it reaches last character.

6.Stop.

copy String ALGORITHM

1.Start.

2.Declare a character array str2 variable.

3.Declare and initialize variable i to zero.

4.Assign ith element of str1 to i th element of str2.

5.increment i by 1.

6.Repeat steps 4 & 5 till length of given string. 7.Assign character ’\0’ to last position of str2.

8.Stop. concatenate ALGO

1.Start.

2.Declare and initialize variable i to zero.

3.Declare a char array varivale res\_str[].

4.Assign character of first string (str1[i]) to res\_str[i]

5.Increment i by 1.

6.Repeat step 4 & 5 till last position of first string.

7.Now, Declare a second variable j and initialize it to zero.

8.assign each character of second string(str2).

9.Increment i and j by 1.

10.Assign char ’\0’ to last position of res\_str[i].

11.Stop.

reverse ALGO

1.Start.

2.Declare and initialize variable i to zero.

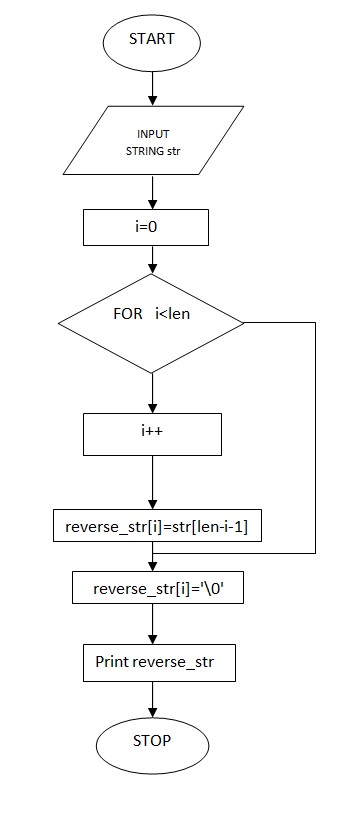
3.Declare a char array varivale res\_str[].

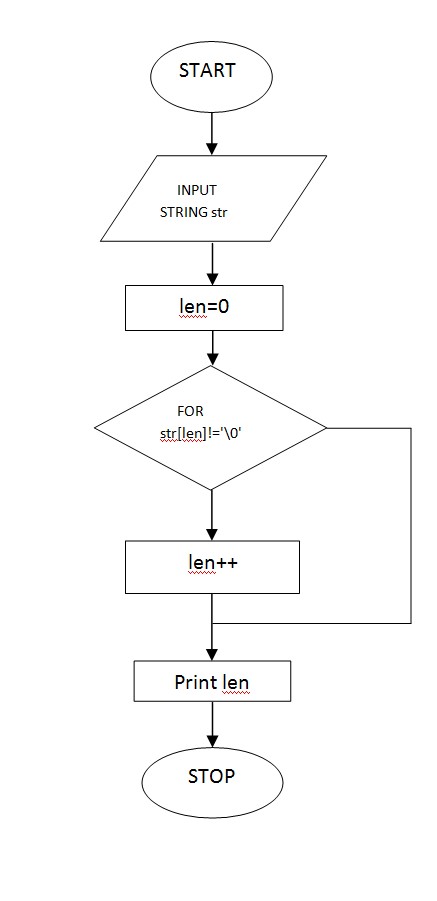
4.Assign each character(Starting from last i.e. (length-1-i)) of given string(str1) to res\_str[i].

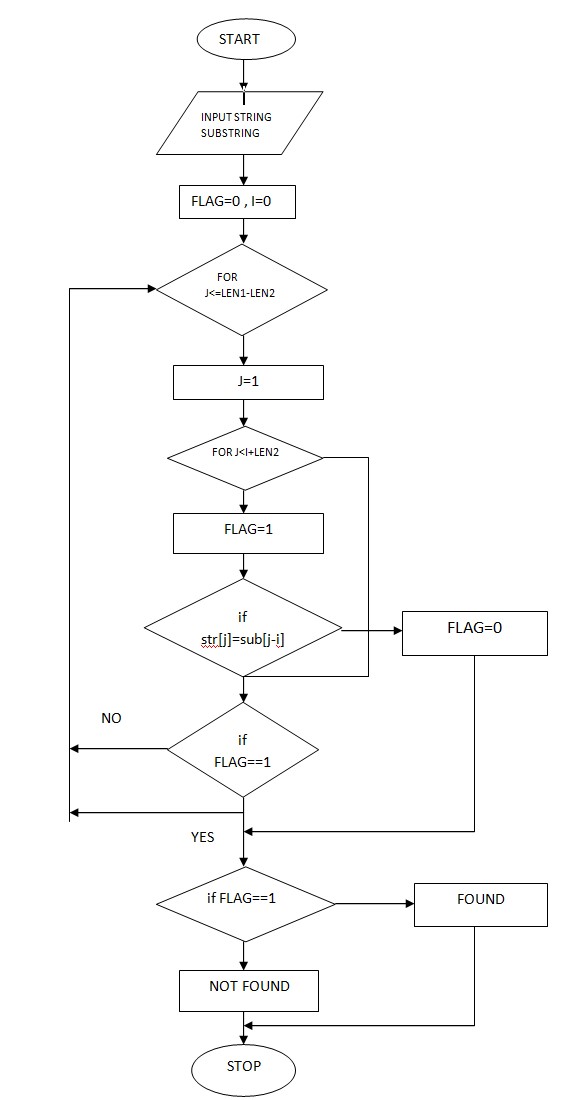
5.Increment i by 1.

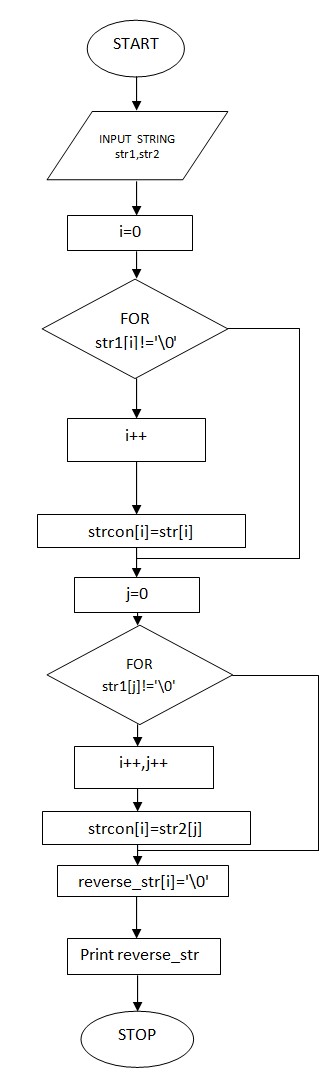
6.Stop.

## Flowchart









Output:

