EDUCBA
(https://www.educba.com/software-development/)

← (https://www.educba.com/escape-sequence-in-c/)

→ (https://www.educba.com/leap-year-program-in-c/)

Introduction to Reverse String in C

Reverse String can be defined as an operation in which the original string which the use is modified in such a way that the characters in it are arranged in a reverse manner starting from the last character to the first character, thus by forming a new string which will be the

characters in a string from end to start and append one after one. This way, we will have a new string formed by reverse traversal, and this string will be the reversed string. In C language, as we don't have support for a string data type, we need to use a character array instead. It is easy here to traverse the character array character by character and form a new character array.

Examples of Reverse String in C

Following are the different examples of reverse string in c using various methods.

Example #1 – Using For Loop

Code:

```
#include <stdio.h>
#include <string.h>
int main ()
{
// char array to take input
char inputString[100];
// char array to build output
char outputString[100];
int length;
int i;
// Take input from the user : input in character array
```

(https://www.educba.com/software-development/)

```
length = strlen( inputString );
int j = 0;
// Traverse character by character from end to start and form a
new string
for( i = length - 1; i >= 0; i--) {
outputString[ j ] = inputString[ i ];
j++;
}
printf( "The reversed string is: ");
printf( "%s", outputString );
printf( "\n" );
return 0;
}
```

Output:

Here, we have used strlen() from the <string.h> library to find out the count of characters present in the input string and passed it in for loop. We have parsed array from end to start and appended characters in reverse order in an output array using for loop.

Example #2 – Using While Loop

Code:

```
#include <stdio.h>
```

(https://www.educba.com/software-development/)

```
char inputString[100];
// char array to build output
char outputString[100];
int length;
int i;
// Take input from the user : input in character array
printf( "Please Enter a string to be reversed \n" );
scanf( "%s", inputString );
// Find the number of characters or length of a string using in
built function strlen() from string.h library
length = strlen( inputString );
int j = 0;
// Traverse character by character from end to start and form a
new string
i = length - 1;
while( i >= 0) {
outputString[ j ] = inputString[ i ];
i--;
j++;
}
printf( "The reversed string is: ");
printf( "%s", outputString );


printf( "\n" );
return 0;
}
```

Output:

Example #3 – Using do While Loop

Let's modify the same code with a do-while loop.

Code:

```
#include <stdio.h>
```

(https://www.educba.com/software-development/)

```
char inputString[100];
// char array to build output
char outputString[100];
int length;
int i;
// Take input from the user : input in character array
printf( "Please Enter a string to be reversed \n" );
scanf( "%s", inputString );
// Find the number of characters or length of a string using in
built function strlen() from string.h library
length = strlen( inputString );
int j = 0;
// Traverse character by character from end to start and form a
new string
i = length - 1;
do {
outputString[ j ] = inputString[ i ];
i--;
j++;
}while( i >= 0);
printf( "The reversed string is: ");
printf( "%s", outputString );


printf( "\n" );
return 0;
}
```

(https://www.educba.com/software-development/)

We can't enter an empty string in input because the C language will not allow it.

Example #4 – Using Swapping

Code:

```
#include <stdio.h>
#include <string.h>
int main ()
{
// char array to take input
char inputString[100];
int length;
int i;
// Take input from the user : input in character array
printf( "Please Enter a string to be reversed \n" );
scanf( "%s", inputString );
// Find the number of characters or length of a string using in
built function strlen() from string.h library
length = strlen( inputString );
// swap characters from start with characters from end
int j = length -1;

char temp;
for( i = 0; i <= (length-1) /2; i++) {
temp = inputString[i];
```

(https://www.educba.com/software-development/)

```
printf( "%s", inputString );
printf( "\n" );
return 0;
}
```

Output:

Here, we are not using any extra character array for storage. We are modifying the existing input character array by swapping the characters from the start with the characters from the end. We need to use only one extra memory space for storage in this case.

Conclusion

String reverse is an operation in which the sequence of characters in a string is reversed. C language provides efficient ways to implement and perform this operation. In this article, we have seen various methods by which we can perform the reverse operation.

Recommended Articles

This is a guide to Reverse String in C. Here we discuss the Introduction along with different examples and code implementation. You may also look at the following articles to learn m

1. Reverse String in Java (https://www.educba.com/reverse-string-in-java/)
2. Reverse String in PHP (https://www.educba.com/reverse-string-in-php/)
3. Reverse String in JavaScript (https://www.educba.com/reverse-string-in-javascript/)

(https://www.educba.com/software-development/)

Terms and Conditions (https://www.educba.com/terms-and-conditions/?source=footer)

Privacy Policy (https://www.educba.com/privacy-policy/?source=footer)

Apps

iPhone & iPad (https://itunes.apple.com/in/app/educba-learning-app/id1341654580?mt=8)

Android (https://play.google.com/store/apps/details?id=com.educba.www)

Resources

Free Courses (https://www.educba.com/software-development/free-courses/?source=footer)

Java Tutorials (https://www.educba.com/software-development/software-development-tutorials/java-tutorial/?source=footer)

Python Tutorials (https://www.educba.com/software-development/software-development-tutorials/python-tutorial/?source=footer)

All Tutorials (https://www.educba.com/software-development/software-development-tutorials/?source=footer)

Certification Courses

All Courses (https://www.educba.com/software-development/courses/?source=footer)

Software Development Course - All in One Bundle (https://www.educba.com/software-development/courses/software-development-course/?source=footer)

Become a Python Developer (https://www.educba.com/software-development/courses/python-certification-course/?source=footer)

(https://www.educba.com/software-development/)

ASP.NET Course (https://www.educba.com/software-development/courses/asp-net-course/?source=footer)

VB.NET Course (https://www.educba.com/software-development/courses/vb-net-course/?source=footer)

PHP Course (https://www.educba.com/software-development/courses/php-course/?source=footer)