Reverse a string in C/C++ using Client Server model

This article describes a Client and Server setup where a Client connects, sends a string to server and the server shows the original string and sends reversed string to client using socket connection.

Prerequisite : Socket Programming

Examples:

```
Input : welcome
Output :emoclew

Input :geeks for geeks
Output :skeeg rof skeeg
```

Explanation

In this, first setup client-server connection. When connection will setup, client will send user input string to server by send system call. At server side, server will wait for string sent by client. Server read string by read system call. After this, server will reverse the string and send back to client.

Compiling :

1. First run the server program as

```
gcc Server.c -o server
```

2. Run the client program on another terminal

```
gcc Client.c -o client
```

3. Server program is waiting for string sent by client.
4. Input the string in client side.
5. Server program will print original string.
6. Client program will print reversed string.

Recommended: Please try your approach on ***{IDE}*** first, before moving on to the solution.

Client.c

C

```c
// C client code to send string to reverse
#include <arpa/inet.h>
#include <netinet/in.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#include <unistd.h>

#define PORT 8090

// Driver code
int main()
{
    struct sockaddr_in address;
    int sock = 0, valread;
    struct sockaddr_in serv_addr;
    char str[100];

    printf("\nInput the string:");
    scanf("%[^\n]s", str);

    char buffer[1024] = { 0 };

    // Creating socket file descriptor
    if ((sock = socket(AF_INET,
```

```
                        SOCK_STREAM, 0))
        < 0) {
        printf("\n Socket creation error \n");
        return -1;
    }

    memset(&serv_addr, '0', sizeof(serv_addr));
    serv_addr.sin_family = AF_INET;
    serv_addr.sin_port = htons(PORT);

    // Convert IPv4 and IPv6 addresses from
    // text to binary form 127.0.0.1 is local
    // host IP address, this address should be
    // your system local host IP address
    if (inet_pton(AF_INET, "127.0.0.1",
                  &serv_addr.sin_addr)
        <= 0) {
        printf("\nAddress not supported \n");
        return -1;
    }

    // connect the socket
    if (connect(sock, (struct sockaddr*)&serv_addr,
                sizeof(serv_addr))
        < 0) {
        printf("\nConnection Failed \n");
        return -1;
    }

    int l = strlen(str);

    // send string to server side
    send(sock, str, sizeof(str), 0);

    // read string sent by server
    valread = read(sock, str, l);

    printf("%s\n", str);

    return 0;
}
```

Server.c

C

```
// Server C code to reverse a
// string by sent from client
#include <netinet/in.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#include <unistd.h>
```

```
#define PORT 8090

// Driver code
int main()
{
    int server_fd, new_socket, valread;
    struct sockaddr_in address;
    char str[100];
    int addrlen = sizeof(address);
    char buffer[1024] = { 0 };
    char* hello = "Hello from server";

    // Creating socket file descriptor
    if ((server_fd = socket(AF_INET,
                            SOCK_STREAM, 0)) == 0) {
        perror("socket failed");
        exit(EXIT_FAILURE);
    }

    address.sin_family = AF_INET;
    address.sin_addr.s_addr = INADDR_ANY;
    address.sin_port = htons(PORT);

    // Forcefully attaching socket to
    // the port 8090
    if (bind(server_fd, (struct sockaddr*)&address,
                            sizeof(address)) < 0) {
        perror("bind failed");
        exit(EXIT_FAILURE);
    }

    // puts the server socket in passive mode
    if (listen(server_fd, 3) < 0) {
        perror("listen");
        exit(EXIT_FAILURE);
    }
    if ((new_socket = accept(server_fd,
                (struct sockaddr*)&address,
                (socklen_t*)&addrlen)) < 0) {
        perror("accept");
        exit(EXIT_FAILURE);
    }

    // read string send by client
    valread = read(new_socket, str,
                   sizeof(str));
    int i, j, temp;
    int l = strlen(str);

    printf("\nString sent by client:%s\n", str);

    // loop to reverse the string
    for (i = 0, j = l - 1; i < j; i++, j--) {
        temp = str[i];
        str[i] = str[j];
        str[j] = temp;
```

```
    }

    // send reversed string to client
    // by send system call
    send(new_socket, str, sizeof(str), 0);
    printf("\nModified string sent to client\n");

    return 0;
}
```

Last Updated : 26 Jan, 2018

2

Previous Next

Article Contributed By :

GeeksforGeeks

Vote for difficulty

Easy Normal Medium Hard Expert

Article Tags : system-programming, C Language

Improve Article Report Issue

GeeksforGeeks
A-143, 9th Floor, Sovereign Corporate Tower,
Sector-136, Noida, Uttar Pradesh - 201305

feedback@geeksforgeeks.org

Company

About Us

Careers

Trending @GfG

Job Fair For Students

GfG Weekly #100

In Media
Contact Us
Terms and Conditions
Privacy Policy
Copyright Policy
Third-Party Copyright Notices
Advertise with us

POTD: Revamped
Python Backend LIVE
Android App Development
DevOps LIVE
DSA in JavaScript

Languages

Python
Java
C++
GoLang
SQL
R Language
Android Tutorial

Data Structures

Array
String
Linked List
Stack
Queue
Tree
Graph

Algorithms

Sorting
Searching
Greedy
Dynamic Programming
Pattern Searching
Recursion
Backtracking

Web Development

HTML
CSS
JavaScript
Bootstrap
ReactJS
AngularJS
NodeJS

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning Tutorial
Maths For Machine Learning
Pandas Tutorial
NumPy Tutorial
NLP Tutorial

Interview Corner

Company Preparation
Preparation for SDE
Company Interview Corner
Experienced Interview
Internship Interview
Competitive Programming
Aptitude

Python

Python Tutorial

GfG School

CBSE Notes for Class 8

Python Programming Examples

Django Tutorial

Python Projects

Python Tkinter

OpenCV Python Tutorial

CBSE Notes for Class 9

CBSE Notes for Class 10

CBSE Notes for Class 11

CBSE Notes for Class 12

English Grammar

UPSC/SSC/BANKING

SSC CGL Syllabus

SBI PO Syllabus

IBPS PO Syllabus

UPSC Ethics Notes

UPSC Economics Notes

UPSC History Notes

Write & Earn

Write an Article

Improve an Article

Pick Topics to Write

Write Interview Experience

Internships

Video Internship

@geeksforgeeks , Some rights reserved