

Table of content:


[Home](#) > [Computer Science](#) > [Advantages And Disadvantages Of Distributed Systems You...](#)

Advantages And Disadvantages Of Distributed Systems You Must Know!




Shivangi Vatsal

Listen



A distributed system is a computing environment where work is distributed among multiple components that collaborate with each other for the purpose of solving a particular problem. Here, the components are separated but they interact with each other and appear as one single system to the end-user.



In the past, the demand for computation was far greater than the technology capabilities at that time. So, for better computation and faster speed, the solution was to make different systems work together in harmony to solve a particular problem. This forms the premise for the development of a special kind of system called the "Distributed System". The first distributed systems were local-area networks (LANs) such as the Ethernet, which was invented in the 1970s. These systems made computation easier due to

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)**Ok, Continue**

that collaborate with each other for the purpose of solving a particular problem. Here, the components are separated but they interact with each other and appear as one single system to the end-user.



• Client-server systems

In this type of system, a client interacts with only one server for requesting a resource. But, a server may interact with many clients at a given instant of time. Thus, the interaction between these different components makes the system a distributed one.

• Peer to peer systems

This type of system has nodes with equal participation and authority. They interact with each other without depending on a single centralized server. The work is equally divided among the peer nodes in a peer-to-peer system.

To help you understand the importance of Distributed systems in our life, let's see some examples of distributed systems.

Examples of Distributed Systems in real life

Domain	Use of Distributed System
Healthcare	Storing and Accessing medicine and patient information
E-Commerce	For storing and accessing payment information
Tracking Systems	Global Positioning System(GPS) for tracking one's location.
Gaming	In Multiplayer Games, allowing different players to play a single game.

Advantages of Distributed Systems

Distributed systems have a myriad of advantages that make their use popular. Few important ones include:

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)



A distributed system doesn't depend on a single node. So, even if there is a single node malfunctioning, other nodes continue to function properly. Thus, the system is intact.

- **Reliability**

For a system to be reliable, it should handle errors efficiently. As distributed systems easily handle system crashes, they are quite reliable.

- **Efficiency**

Distributed systems are highly efficient as they involve multiple computers that save time for users. Also, they can provide higher performance as compared to centralized systems.

- **Lesser delay**

In today's world, time is an important constraint for business personnel and users alike. Distributed Systems provide a low latency rate. For example, consider a user who uses the internet and loads a website. The system makes sure that the node located closer to the user is used to perform the loading task in order to save time.

Distributed Systems	
Advantages	Disadvantages
<ul style="list-style-type: none">• Easy data sharing• Unmatched scalability• Good failure handling• Reliable	<ul style="list-style-type: none">• Security issue• High set-up cost• Data loss• Overloading issue

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

Disadvantages of Distributed Systems



Security issues usually occur in many software and hardware devices. The same case is with Distributed Systems. Such security risks occur as a result of many nodes and connections in an open system setting that makes it difficult to ensure adequate security.

- **High set-up cost**

The initial cost of installation and set-up is high due to many hardware and software devices. There are other maintenance costs associated with the system which adds to the total cost, making it even more expensive.

- **Data loss**

There can be instances when the data sent from one node to another node can be lost midway in its journey from the source node to the destination node. This results in some important messages being lost, which is detrimental to a company.

- **Difficult to handle**

The hardware and software of a distributed system are quite complex. It's complicated to maintain and operate the hardware components for companies. Also, software complexity makes it necessary for companies to pay special attention to the software components.

- **Overloading issue**

The Overloading issue can occur in the system if all the nodes of the distributed system try to send data at one particular instant of time.

In a nutshell, we can say that distributed systems have a significant impact on our lives. But, there are some issues that occur while using them, mostly regarding security and complexity.

To further brush up on your basics, read:

- [Guide 101: DBMS advantages and disadvantages, applications and top resources for preparation](#)

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

Shivangi Vatsal

Sr. Associate Content Strategist @Unstop





Tags:

Information Technology

Engineering

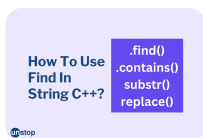
Data Science and Machine Learning

Comments

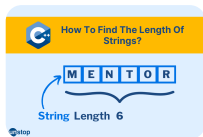
Frequently Asked Questions (FAQs) and Discussions.

Please [login](#) to start a comment.

Blogs you need to hog!



Find In Strings C++ | Examples To Find Substrings, Character & More!



Ways To Find String Length In C++ Simplified (With Examples)



Strings In C++ | Functions, How To Convert & More (With Examples)



What Is C++? An Introduction To C++ Programming Like No Other!



~

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

Find In Strings C++ | Examples To Find Substrings, Character & More!



How To Use Find In String C++?

`.find()`
`.contains()`
`substr()`
`replace()`



Table of content:



In programming, strings are like sentences that consist of a series of characters and are used to store information. There are different kinds of [strings in C++](#) and we can fulfill a variety of tasks with the help of strings and substring functions. In this article, we will have a look at what are substrings in C++, along with ways to find strings in C++, find substrings, partial strings, a single character of the string, etc. We have provided detailed examples for each to help you better understand these topics.

What Is Find In String C++?

In simple terms, the `find()` function is the method that returns the index of the first occurrence of a

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

The syntax for declaring `.find()` function is-



Here,

str1- the input string/ main string/ original string

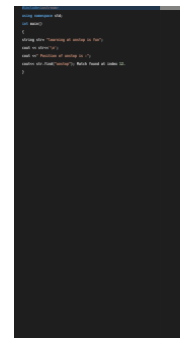
substring- the string to be matched to the main/ original string

Return value

The function finds the substring that matches str1 and returns the position at which the substring is found. In case, the substring passed as an argument, does not match the input string, a special [variable](#) called `std::string::npos` is returned to denote the end of the string.

Implementation example of .find() function-

```
1  #include<iostream>
2
3  using namespace std;
4
5  int main()
6  {
7
8
9  string str= "learning at unstop is fun";
10
```



Output-

learning at unstop is fun

Position of unstop is 12

Explanation-

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

answer? No, let's get into the depth a little more. A substring is a part of a string which is continuous and can be repetitive as well. It could be in the beginning of the original string, anywhere in the middle of



To find a substring in a string, we have a predefined function called `substr()` function in our C++ programming language. The `substr()` function takes 2 parameters `pos` and `len` as arguments to find and return the substring between the specified position and length.

Syntax of Substr() function:

```
// string substr (size_t pos, size_t len) const;
```

Knowing the parameters:

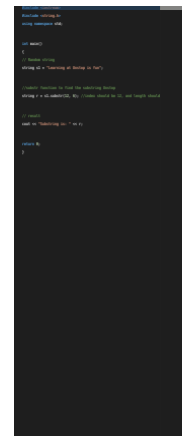
`size_t` : is an unsigned integral type

`pos`: starting position/index from where string search begins

`len`: length of the substring

Example for implementation of substr() function:

```
1  #include <iostream>
2
3  #include <string.h>
4
5  using namespace std;
6
7
8
9
10 int main()
11 {
12
```



Output

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

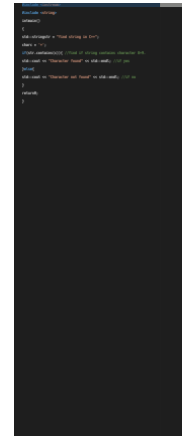
use the `str.substring()` function to find a substring with index as 12 and the length of the string as 6. Once, the substring is found, we store it in a string `r` and print it as our output.



To find a specific/ single character in a string in C++, we can use the `std::basic_string::contains` function. The return value of this function is boolean, which means it return true is the character is found else false.

Implementation of `.contains()` function:

```
1  #include <iostream>
2
3  #include <string>
4
5  intmain()
6  {
7
8
9  std::stringstr = "find string in C++";
10
11  charc = '+';
12
```



Output:

Character Found

Explanation:

In the above example, we initialize `str` - string with a random string and `char c` with a random character. We then make use of the `str.contains()` function in C++, to find if the random string contains the input character. If yes then 'character found' is printed, otherwise 'character not found' is printed.

Also read- [Difference Between C And C++ | Features | Application & More!](#)

Find All Substrings From A Given String In C++

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

```
2
3
4  #include<bits/stdc++.h>
```





```
10
11 // Function to print all sub strings
12
13 void subString(string s, int n)
14
15 {
16
17     for (int i = 0; i < s.length(); i++)
```

Output-

r

ru

run

u

un

n

Explanation:

In the above example, the subString functions print all substrings of the original string. The outer loop goes to the [length of the entire string](#), in this case, 3. The inner loop prints individual substrings of all lengths until (n-i), in this case, substrings of lengths 1,2, and 3. The main function initializes a string s as run. This main function drives the subString function which in turn prints all the substring in a given string in C++.

Index Substring In String In C++ From A Specific Start To A Specific Length

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

Here,





Implementation to index substring in a string from specified start to end:

```
1  # include <iostream>
2
3  # include <string>
4
5  using namespace std;
6
7  int main ()
8
9  {
10
11  //declare string and substring
12
```



Output-

s: find string in C++

subs: string

Explanation:

In the above example, we first declare a string `s` and a substring `subs`. We then initialize start and end index for the substring as 5 and 11. Then, the function `substr()` is used to access 6 characters from 5th index hence, $(\text{end} - \text{start}) = 11 - 5 = 6$. This gives us our substring which we print as output along with our main string.

Conclusion

In this article, we discussed all the way to find strings in C++. So, we are familiar with the fact that strings are just normal text that we type or read in our day-to-day life, and in programming language we

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

Q. How do I find the substring of a string in C++?



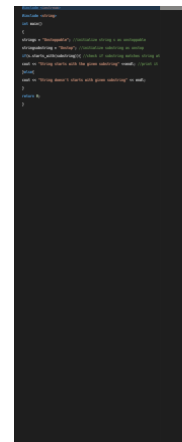
Syntax of Substr() function:

```
// string substr (size_t pos, size_t len) const;
```

Q. How do you check if a string starts with a specific substring?

string::starts_with function is used to find if a string starts with a specified substring. This function returns true if the string starts with a specified substring, else false. It can be illustrated as:

```
1  #include <iostream>
2
3  #include <string>
4
5  int main()
6  {
7
8
9  strings = "Unstoppable"; //initialize string s as unstoppable
10
11 stringsubstring = "Unstop"; //initialize substring as unstop
12
```



Output:

String starts with the given prefix

Q. How do you find a substring in a string and replace it in C++?

C++ string library provides a function called string replace() function, to replace a portion of string starting from a given index to the end index and replacing that with the new string. The position from where the substring has to be started, the length of the substring, and the new string is passed in the parameter as an argument.

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

C++ library provides a .find() function which returns the index of the first occurrence of substring in string. If case when no substring is found, string::npos is returned.

Q. What is the return value of find_first_of in C++?

The function find_first_of in C++ returns the position of any element from range[first1,last1] that matches with the elements in the range[first2,last2]. In case, there are more than one element that matches both the range, then the position of the first common element is returned. When there is no match found in the entire range, the function returns string :: npos.

Q. What is the use of find_last_of function in C++?

The function find_last_of is used to find the index of the last occurrence of a character in a string. If the character is present in the string, it returns the index of the last occurrence of that character in the string; else it returns string::npos.

We have reached the end of the article on using 'find' in strings in C++. You might also be interested in reading:

- [Typedef In C++ | Syntax, Application & How To Use It \(With Examples\)](#)
- [What Is GitHub? An Introduction, How-To Use It, Components & More!](#)
- [History Of C++ | Detailed Explanation \(With Timeline Infographic\)](#)
- [The Structure of C++ Programs Explained With Examples!](#)
- [What Is Bash? Features, Major Concepts, Commands, & More!](#)

Edited by



Shivani Goyal

Sr. Associate Content

I am an economics graduate using my qualifications and life skills to observe & absorb what life has to offer. A strong believer in 'Don't die before you are dead' philosophy, at Unstop I am producing content that resonates and enables you to be #Unstoppable. When I don't have to be presentable for the job, I'd be elbow deep in paint/ pencil residue, immersed in a good read or socializing in the flesh.

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)



[Learn](#)[Practice](#)[Mentorship](#)[Compete](#)[Jobs](#)[Host](#)[Login](#)

We serve cookies to deliver our services, personalise information and improve your experience. By using Unstop, you agree to our [Privacy Policy](#) and [Terms of Use](#).

[Privacy Policy](#) [Terms of Use](#)

