GeeksforGeeks

Array of Strings in C++ (5 Different Ways to Create)

Last Updated: 28-10-2020

In C and C++, a string is a 1-dimensional array of characters and an array of strings in C is a 2-dimensional array of characters. There are many ways to declare them, and a selection of useful ways are given here.

1. Using Pointers:

We actually create an array of string literals by creating an array of pointers.

This is supported by both C and C++.

CPP

```cpp
// C++ program to demonstrate array of strings using
// 2D character array
#include <iostream>

int main()
{
    // Initialize array of pointer
    const char *colour[4] = { "Blue", "Red",
                              "Orange", "Yellow" };

    // Printing Strings stored in 2D array
    for (int i = 0; i < 4; i++)
        std::cout << colour[i] << "\n";

    return 0;
}
```

Output:

```
Blue
Red
Orange
Yellow
```

- Number of strings are fixed, but needn't be. The 4 may be omitted, and the compiler will compute the correct size.
- These strings are constants and their contents cannot be changed. Because string literals (literally, the quoted strings) exist in a read-only area of memory, we must specify "const" here to prevent unwanted accesses that may crash the program.

2. Using 2D array:

This method is useful when the length of all strings is known and a particular memory footprint is desired. Space for strings will be allocated in a single block

This is supported in both C and C++.

CPP

```
// C++ program to demonstrate array of strings using
// 2D character array
#include <iostream>

int main()
{
    // Initialize 2D array
    char colour[4][10] = { "Blue", "Red", "Orange",
                           "Yellow" };
```

Output:

```
Blue
Red
Orange
Yellow
```

- Both the number of strings and size of strings are fixed. The 4, again, may be left out, and the appropriate size will be computed by the compiler. The second dimension, however, must be given (in this case, 10), so that the compiler can choose an appropriate memory layout.
- Each string can be modified, but will take up the full space given by the second dimension. Each will be laid out next to each other in memory, and can't change size.
- Sometimes, control over the memory footprint is desirable, and this will allocate a region of memory with a fixed, regular layout.

3. Using the string class:

The STL string class may be used to create an array of mutable strings. In this method, the size of the string is not fixed, and the strings can be changed.

This is supported only in C++, as C does not have classes.

CPP

```cpp
// C++ program to demonstrate array of strings using
// array of strings.
#include <iostream>
#include <string>

int main()
{
    // Initialize String Array
```

Output:

```
Blue
Red
Orange
Yellow
```

- The array is of fixed size, but needn't be. Again, the 4 here may be omitted, and the compiler will determine the appropriate size of the array. The strings are also mutable, allowing them to be changed.

4. Using the vector class:

The STL container Vector can be used to dynamically allocate an array that can vary in size.

This is only usable in C++, as C does not have classes. Note that the initializer-list syntax here requires a compiler that supports the 2011 C++ standard, and though it is quite likely your compiler does, it is something to be aware of.

CPP

```cpp
// C++ program to demonstrate vector of strings using
#include <iostream>
#include <vector>
#include <string>

int main()
{
    // Declaring Vector of String type
    // Values can be added here using initializer-list syntax
    std::vector<std::string> colour {"Blue", "Red", "Orange"};

    // Strings can be added at any time with push_back
    colour.push_back("Yellow");
```

Output:

```
Blue
Red
Orange
Yellow
```

- Vectors are dynamic arrays, and allow you to add and remove items at any time.
- Any type or class may be used in vectors, but a given vector can only hold one type.

5. Using the array class:

The STL container array can be use to allocate a fixed-size array. It may be used very similarly to vector, but the size is always fixed.

This is supported only in C++.

C++

```
    std::array<std::string, 4> colour { "Blue", "Red", "Orange",
                                     "Yellow" };

    // Printing Strings stored in array
    for (int i = 0; i < 4; i++)
        std::cout << colour[i] << "\n";

    return 0;
}
```

Output

```
Blue
Red
Orange
Yellow
```

Notes:

These are by no means the only ways to make a collection of strings. C++ offers several container classes, each of which have various tradeoffs and features, and all of them exist to fill requirements that you will have in your projects. Explore and have fun!

Conclusion: Out of all the methods, Vector seems to be the best way for creating an array of Strings in C++.

This article is contributed by **Kartik Ahuja**. If you like GeeksforGeeks and would like to contribute, you can also write an article using contribute.geeksforgeeks.org or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Attention reader! Don’t stop learning now. Get hold of all the important DSA concepts with the **DSA Self Paced Course** at a student-friendly price and become industry ready.

Recommended Posts:

Different ways for Integer to String Conversions In Java

Different ways of Method Overloading in Java

Different ways of Reading a text file in Java

Different ways to delete elements in std::map (erase() and clear())

Different ways to declare variable as constant in C and C++

Initialize a vector in C++ (5 different ways)

Preventing Object Copy in C++ (3 Different Ways)

Ways to express a number as product of two different factors

Reverse string in Python (5 different ways)

Print system time in C++ (3 different ways)

Different ways to initialize a variable in C/C++

Different ways to use Const with Reference to a Pointer in C++

Different Ways to Remove all the Digits from String in Java

Different Ways to Generate String by using Characters and Numbers in Java

8 different ways to Add Two Numbers in C/C++

Print all Strings from array A[] having all strings from array B[] as subsequence

Minimum cost to sort strings using reversal operations of different costs

Different substrings in a string that start and end with given strings

Program to build a DFA that accepts strings starting and ending with different character

Search in an array of strings where non-empty strings are sorted

55

1.7

□ To-do □ Done

Based on **66** vote(s)

Improve Article

Please write to us at contribute@geeksforgeeks.org to report any issue with the above content.

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments

GeeksforGeeks

5th Floor, A-118,
Sector-136, Noida, Uttar Pradesh - 201305

feedback@geeksforgeeks.org

Company

About Us

Careers

Learn

Algorithms

Data Structures

GeeksforGeeks

Practice

Courses

Company-wise

Topic-wise

How to begin?

Contribute

Write an Article

Write Interview Experience

Internships

Videos