

A Puzzle on
C/C++ R-Value
Expressions

Templates and Static variables in C++

Template
Specialization
in C++

Templates in
C++

Trie Data
Structure
using smart
pointer and
OOP in C++

C++ program
to compare
two Strings

→ Using Operator
Overloading

Derived Data
Types in C++

Go vs C++

Pointers and
References in
C++

fill in C++ STL

Deque vs
Vector in C++

STL

Types of
Literals in
C/C++ with
Examples

Strings in C++
and How to
Create them?

Conditional or
Ternary
Operator (?:) in
C/C++

C++
Programming
Basics

not1 and not2
function
templates in
C++ STL with
Examples

C++ | asm
declaration

→ Enum Classes
in C++ and
Their
Advantage
over Enum
DataType

How to Insert
an element at
a specific
position in an
Array in C++

C++ Program
to print an
Array using

Recursion

Count of
distinct
remainders
when N is
divided by all
the numbers
from the range
[1, N]

Difference
Between
Constructor
and Destructor
in C++

How to erase
an element
from a vector
using erase()
and
reverse_iterator?

How to
implement our
own Vector
Class in C++?

→ Difference
between
Inheritance
and
Polymorphism

Introduction to
C++
Programming
Language

OpenCV |
Hands on
Image
Contrast

ios eof()
function in
C++ with
Examples

Difference
between
Abstraction
and
Encapsulation
in C++

return
statement in
C/C++ with
Examples

Hmm. We
having
trouble
finding th
site.

We can't connect to t
server at
googleads.g.doublecl

**If that address is cor
here are three other
you can try:**

- Try again later.
- Check your netw
connection.
- If you are connec
but behind a fire
check that Firefo
permission to ac
the Web.

Templates and Static variables in C++

Function templates and static variables:

Each instantiation of function template has its own copy of local static variables. For example, in the following program there are two instances: *void fun(int)* and *void fun(double)*. So two copies of static variable *i* exist.

```
#include <iostream>

using namespace std;

template <typename T>
void fun(const T& x)
{
    static int i = 10;
    cout << ++i;
    return;
}

int main()
{
    fun<int>(1); // prints 11
    cout << endl;
    fun<int>(2); // prints 12
    cout << endl;
    fun<double>(1.1); // prints 11
    cout << endl;
    getch();
    return 0;
}
```

Output of the above program is:

```
11
12
11
```

Class templates and static variables:

The rule for class templates is same as function templates

Each instantiation of class template has its own copy of member static variables. For example, in the following program there are two instances *Test* and *Test*. So two copies of static variable *count* exist.

Hmm. We're having trouble finding that site.

We can't connect to the server at googleads.g.doubleclick.net.

If that address is correct, here are three other things you can try:

- Try again later.
- Check your network connection.

```
#include <iostream>

using namespace std;

template <class T> class Test
{
private:
    T val;
public:
    static int count;
    Test()
    {
        count++;
    }
    // some other stuff in class
};

template<class T>
int Test<T>::count = 0;

int main()
{
    Test<int> a; // value of count for Test<int> is 1 now
    Test<int> b; // value of count for Test<int> is 2 now
    Test<double> c; // value of count for Test<double> is 1 now
    cout << Test<int>::count << endl; // prints 2
    cout << Test<double>::count << endl; //prints 1

    getchar();
    return 0;
}
```

Output of the above program is:

```
2
1
```

Please write comments if you find anything incorrect, or you want to share more information

about the topic discussed above.

Hmm. We're having trouble finding that site.

We can't connect to the server at googleads.g.doubleclick.net.

Recommended Posts:

Can static functions be virtual in C++?

Some interesting facts about static member functions in C++

Comparison of static keyword in C++ and Java

Templates and Default Arguments

Static data members in C++

When are static objects destroyed?

Swap two variables in one line in C/C++, Python, PHP and Java

Variadic function templates in C++

Templates in C++

Scope of Variables in C++

Static Keyword in C++

Templates in C++ vs Generics in Java

Count the number of objects using Static member function

Can Global Variables be dangerous ?

→ "static const" vs "#define" vs "enum"

Article Tags : C++

Practice Tags : CPP



3

2.6

☐ To-do ☐ Done

Based on 29 vote(s)

[Feedback/ Suggest Improvement](#)[Add Notes](#)[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
A computer science portal for geeks

5th Floor, A-118,
Sector-136, Noida, Uttar Pradesh - 201305
feedback@geeksforgeeks.org

COMPANY

About Us
Careers
Privacy Policy
Contact Us

LEARN

Algorithms
Data Structures
Languages
CS Subjects
Video Tutorials

PRACTICE

Courses
Company-wise
Topic-wise
How to begin?

CONTRIBUTE

Write an Article
Write Interview Experience
Internships
Videos

@geeksforgeeks, Some rights reserved

