Macros and its types in C/C++

Difficulty Level : Basic   ●   Last Updated : 19 Oct, 2021

A **macro** is a piece of code in a program that is replaced by the value of the macro. Macro is defined by **#define** directive. Whenever a macro name is encountered by the compiler, it replaces the name with the definition of the macro. Macro definitions need not be terminated by semi-colon(**;**).

Below are the program to illustrate the use of macros in C/C++:

Program 1:

C   ▼

```c
// C program to illustrate macros
#include <stdio.h>

// Macro definition
#define LIMIT 5

// Driver Code
int main()
{
    // Print the value of macro defined
    printf("The value of LIMIT"
           " is %d",
           LIMIT);

    return 0;
}
```

C++

Start Your Coding Journey Now!

```cpp
    // Print the value of macro defined
    cout << "The value of LIMIT"
         << " is " << LIMIT;

    return 0;
}
```

Output:

```
 The value of LIMIT is 5
```

Program 2:

C   ▾

```c
// C program to illustrate macros
#include <stdio.h>

// Macro definition
#define AREA(l, b) (l * b)

// Driver Code
int main()
{
    // Given lengths l1 and l2
    int l1 = 10, l2 = 5, area;

    // Find the area using macros
    area = AREA(l1, l2);

    // Print the area
    printf("Area of rectangle"
           " is: %d",
           area);

    return 0;
}
```

Start Your Coding Journey Now!

```cpp
// Driver Code
int main()
{
    // Given lengths l1 and l2
    int l1 = 10, l2 = 5, area;

    // Find the area using macros
    area = AREA(l1, l2);

    // Print the area
    cout << "Area of rectangle"
        << " is: ",
        area;

    return 0;
}
```

Output:

```
Area of rectangle is: 50
```

Explanation:
From the above program we can see that whenever the compiler finds **AREA(l, b)** in the program it replaces it with the macros definition i.e., **(l*b)**. The values passed to the macro template **AREA(l, b)** will also be replaced by the statement **(l*b)**. Therefore, **AREA(10, 5)** will be equal to **10*5**.

Types Of Macros

Start Your Coding Journey Now!   Login   Register

1. **Object-like Macros:** An object-like macro is a simple identifier which will be replaced by a code fragment. It is called object-like because it looks like an object in code that uses it. It is popularly used to replace a symbolic name to numerical/variable represented as constant.
   Below is the illustration of a simple macro:

C

```c
// C program to illustrate macros
#include <stdio.h>

// Macro definition
#define DATE 31

// Driver Code
int main()
{
    // Print the message
    printf("Lockdown will be extended"
           " upto %d-MAY-2020",
           DATE);

    return 0;
}
```

C++

Got It !

Start Your Coding Journey Now! | Login | | Register |

```cpp
    // Print the message
    cout << "Lockdown will be extended"
         << " upto " << DATE
         << "-MAY-2020";

    return 0;
}
```

Output:

```
 Lockdown will be extended upto 31-MAY-2020
```

2. **Chain Macros:** Macros inside macros are termed as chain macros. In chain macros first of all parent macro is expand then child macro is expanded.

Below is the illustration of a Chain Macro:

C++

```cpp
// C program to illustrate macros
#include <stdio.h>

// Macro definition
#define INSTAGRAM FOLLOWERS
#define FOLLOWERS 138

// Driver Code
int main()
{
    // Print the message
    printf("Geeks for Geeks have %dK"
           " followers on Instagram",
           INSTAGRAM);

    return 0;
}
```

Start Your Coding Journey Now!

```cpp
#define FOLLOWERS 138

// Driver Code
int main()
{
    // Print the message
    cout << "Geeks for Geeks have "
        << INSTAGRAM << "K followers on Instagram!";

    return 0;
}
```

Output:

Data Structures   Algorithms   Interview Preparation   Topic-wise Practice   C++   Java   Pytho

1. **Explanation:**
   INSTAGRAM is expanded first to produce **FOLLOWERS**. Then the expanded macro is expanded to produce the outtcome as **138**. This is called the chaining of macros.

2. **Multi-line Macros:** An object-like macro could have a multi-line. So to create a multi-line macro you have to use backslash-newline.
   Below is the illustration of multiline macros:

C

```c
// C program to illustrate macros
#include <stdio.h>

// Multi-line Macro definition
#define ELE 1, \
            2, \
            3
```

Start Your Coding Journey Now!

```cpp
    for (int i = 0; i < 3; i++) {
        printf("%d  ", arr[i]);
    }
    return 0;
}
```

C++ ▼

```cpp
// C++ program to illustrate macros
#include <iostream>
using namespace std;

// Multi-line Macro definition
#define ELE 1, \
            2, \
            3

// Driver Code
int main()
{
    // Array arr[] with elements
    // defined in macros
    int arr[] = { ELE };

    // Print elements
    printf("Elements of Array are:\n");

    for (int i = 0; i < 3; i++) {
        cout << arr[i] << ' ';
    }

    return 0;
}
```

Output:

```
Elements of Array are:
```

Start Your Coding Journey Now!

A function-like macro is only lengthened if an only if its name appears with a pair of parentheses after it. If we don't do this, the function pointer will get the address of the real function and lead to a syntax error.
Below is the illustration of function-like macros:

C

```c
// C program to illustrate macros
#include <stdio.h>

// Function-like Macro definition
#define min(a, b) (((a) < (b)) ? (a) : (b))

// Driver Code
int main()
{

    // Given two number a and b
    int a = 18;
    int b = 76;

    printf("Minimum value between"
           " %d and %d is %d\n",
           a, b, min(a, b));

    return 0;
}
```

C++

```cpp
1  // C++ program to illustrate macros
2  #include <iostream>
3  using namespace std;
4
5  // Function-like Macro definition
```

Start Your Coding Journey Now!　　　[Login]　　[Register]

```cpp
13        int a = 18;
14        int b = 76;
15
16        cout << "Minimum value between"
17            << a << " and " << b
18            << " is: " << min(a, b);
19
20        return 0;
21    }
22
```

```
Minimum value between18 and 76 is: 18


https://ide.geeksforgeeks.org/Y30LzTfGxK
```

Output:

```
Minimum value between 18 and 76 is 18
```

Take a step-up from those "Hello World" programs. Learn to implement data structures like Heap, Stacks, Linked List and many more! Check out our **Data Structures in C** course to start learning today.

Start Your Coding Journey Now!

<div style="text-align:right">

Login   Register

</div>

Like   54

Previous   Next

RECOMMENDED ARTICLES

Page : **1** 2 3

01 Data Type Ranges and their macros in C++
05, Jul 16

02 Interesting Facts about Macros and Preprocessors in C
15, Feb 14

03 Output of the Program | Use Macros Carefully!

05 Variable length arguments for Macros
17, Jul 12

06 Branch prediction macros in GCC
01, Nov 12

07 Macros vs Functions
19, Feb 16

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our Cookie Policy & Privacy Policy

Got It !

Start Your Coding Journey Now!

Article Contributed By :

divya_dashrath_barvekar
@divya_dashrath_barvekar

Vote for difficulty

Current difficulty : <u>Basic</u>

| Easy | Normal | Medium | Hard | Expert |

Improved By :   prashansaagarwal101,   anikaseth98,   regishgeorge2000

Article Tags :   C Macro,   cpp-macros,   C Language,   C++

Practice Tags :   CPP

Improve Article   Report Issue

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

Load Comments

5th Floor, A-118,

Start Your Coding Journey Now!

About Us

Careers

Privacy Policy

Contact Us

Copyright Policy

Algorithms

Data Structures

Languages

CS Subjects

Video Tutorials

Web Development

Web Tutorials

HTML

CSS

JavaScript

Bootstrap

Contribute

Write an Article

Write Interview Experience

Internships

Videos

@geeksforgeeks , Some rights reserved