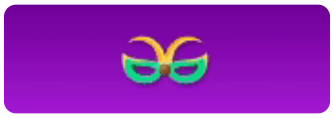
# accumulate() and partial_sum() in C++ STL : numeric header

This header is part of the numeric library in . This article explains some useful functions in the numeric header which can be used during competitive programming to save time and effort.



We usually find out the sum of elements in a particular range or a complete array using a linear operation which requires adding all the elements in the range one by one and storing it into some variable after each iteration.



This function returns the sum of all the values lying in a range between **[first, last)** with the variable sum.

1. **Syntax 1:**

```
accumulate(first, last, sum);
first, last : first and last elements of range
              whose elements are to be added
sum :  initial value of the sum
```

2. **Syntax 2:** This function returns the sum of all the values lying between [first, last) with the variable sum.

```
accumulate(first, last, sum, myfun);
myfun : a function for performing any
        specific task. For example, we can
        find product of elements between
        first and last.
```

```
// C++ program to demonstrate working of accumulate()
#include <iostream>
#include <numeric>
using namespace std;

// User defined function
int myfun(int x, int y)
{
    // for this example we have taken product
    // of adjacent numbers
    return x * y ;
}

int main()
{
    // Initialize sum = 1
    int sum = 1;
    int a[] = {5 , 10 , 15} ;

    // Simple default accumulate function
    cout << "\nResult using accumulate: ";
    cout << accumulate(a , a+3 , sum);

    // Using accumulate function with
    // defined function
    cout << "\nResult using accumulate with"
             "user-defined function: ";
    cout << accumulate(a, a+3, sum, myfun);

    // Using accumulate function with
    // pre-defined function
    cout << "\nResult using accumulate with "
            "pre-defined function: ";
    cout << accumulate(a, a+3, sum, std::minus<int>());

    return 0;
}
```

Output:

```
Result using accumulate: 31
Result using accumulate with user-defined function: 750
Result using accumulate with pre-defined function: -29
```

An Example Problem : Sum of all elements between k1'th and k2'th smallest elements

**partial_sum( )**

This function assigns partial sum of the corresponding elements of an array to every position of the second array.It returns the partial sum of all the set of values lying between **[first, last)** and stores it in

another array b.

For example, if x represents an element in [first, last) and y represents an element in result, the ys can be calculated as:

```
y0 = x0
y1 = x0 + x1
y2 = x0 + x1 + x2
y3 = x0 + x1 + x2 + x3
y4 = x0 + x1 + x2 + x3 + x4
```

Syntax :

```
partial_sum(first, last, b);
partial_sum(first, last, b, myfun);
first, last : first and last element of range
              whose elements are to be added
b : index of array where  corresponding partial
    sum will be stored;
myfun : a user defined function for performing
        any specific task
```

```
// C++ program to demonstrate working of accumulate()
#include <iostream>
#include <numeric>
using namespace std;

//user defined function
int myfun(int x, int y)
{
    // the sum of element is twice of its
    // adjacent element
    return x + 2 * y;
}

int main ()
{
    int a[] = {1, 2, 3, 4, 5} ;
    int b[5];

    // Default function
    partial_sum(a , a+5 , b);

    cout << "Partial Sum - Using Default function: ";
    for (int i=0; i<5; i++)
        cout << b[i] << ' ';
    cout << '\n';

    // Using user defined function
    partial_sum(a , a+5 , b , myfun) ;

    cout << "Partial sum - Using user defined function: ";
    for (int i=0; i<5; i++)
        cout << b[i] << ' ';
    cout << '\n';
```

```
    return 0;
}
```

Output :

```
Partial Sum - Using Default function: 1 3 6 10 15
Partial sum - Using user defined function: 1 5 11 19 29
```

This article is contributed by **Abhinav Tiwari** .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.

## Recommended Posts:

numeric header in C++ STL | Set 2 (adjacent_difference(), inner_product() and iota())
How to write your own header file in C?
clocale header file in C++
Comment in header file name?
random header in C++ | Set 1(Generators)
random header | Set 2 (Distributions)
random header in C++ | Set 3 (Distributions)
Print "Hello World" in C/C++ without using any header file
<complex.h> header file in C with Examples
time.h header file in C with Examples
Difference between Header file and Library
What's difference between header files "stdio.h" and "stdlib.h" ?
Namespace in C++ | Set 3 (Accessing, creating header, nesting and aliasing)

<numeric> library in C++ STL

Split numeric, alphabetic and special symbols from a String

**Article Tags :** C C++ CPP-Library cpp-numerics-library STL

**Practice Tags :** STL C CPP

5

2.5

To-do Done

Based on **10** vote(s)

Feedback/ Suggest Improvement 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