

std::partial_sum

Эта страница была переведена автоматически с английской версии вики используя Переводчик Google (<http://translate.google.com>) . Перевод может содержать ошибки и странные формулировки.



Наведите курсор на текст, чтобы увидеть оригинал. Щёлкните здесь, чтобы увидеть английскую версию этой страницы.

(Вы можете помочь в исправлении ошибок и улучшении перевода. Для инструкций перейдите по ссылке (<http://en.cppreference.com/w/Cppreference:MachineTranslations>) .)

Определено в заголовочном файле `<numeric>`

```
template< class InputIt, class OutputIt >
OutputIt partial_sum( InputIt first, InputIt last, OutputIt d_first ); (1)
```

```
template< class InputIt, class OutputIt, class BinaryOperation >
OutputIt partial_sum( InputIt first, InputIt last, OutputIt d_first,
    BinaryOperation op ); (2)
```

Вычисляет частичных сумм элементов в поддиапазонах диапазона `[first, last)` и записывает их в диапазоне начиная с `d_first`. Первый вариант используется оператор `+` подвести итоги элементы, вторая версия использует данную функцию двоичного `op`. Эквивалентная операция

```
*(d_first) = *first;
*(d_first+1) = *first + *(first+1);
*(d_first+2) = *first + *(first+1) + *(first+2);
*(d_first+3) = *first + *(first+1) + *(first+2) + *(first+3);
...
```

Параметры

- first, last** — диапазон элементов в сумме
- d_first** — В начале назначения диапазона
- op** — binary operation function object that will be applied.

The signature of the function should be equivalent to the following:

```
Ret fun(const Type1 &a, const Type2 &b);
```

The signature does not need to have `const &`.

Тип `Type1` должен быть таков, что объект типа `iterator_traits<InputIt>::value_type` может быть неявно преобразован в `Type1`. Тип `Type2` должен быть таков, что объект типа `InputIt` может быть разыменован и затем неявно преобразован в `Type2`. Тип `Ret` должен быть таков, что объекту типа `iterator_traits<InputIt>::value_type` можно присвоить значение типа `Ret`.

Требования к типам

- `InputIt` должен соответствовать требованиям `InputIterator`.
- `OutputIt` должен соответствовать требованиям `OutputIterator`.

Возвращаемое значение

Итератор на элемент после последнего элемента написано.

Сложность

Именно $(last - first) - 1$ применения бинарной операции

Возможная реализация

Первый вариант

```
template<class InputIt, class OutputIt>
OutputIt partial_sum(InputIt first, InputIt last,
                    OutputIt d_first)
{
    return std::partial_sum(first, last, d_first,
                           std::plus<InputIt, InputIt>());
}
```

Второй вариант

```
template<class InputIt, class OutputIt, class BinaryOperator>
OutputIt partial_sum(InputIt first, InputIt last,
                    OutputIt d_first, BinaryOperation op)
{
    if (first == last) return d_first;

    typename std::iterator_traits<InputIt>::value_type sum = *first;
    *d_first = sum;

    while (++first != last) {
        sum = op(sum, *first);
        *++d_first = sum;
    }
    return ++d_first;
}
```

Пример

Запустить этот код

```
#include <numeric>
#include <vector>
#include <iostream>
#include <iterator>
#include <functional>
int main()
{
    std::vector<int> v = {2,2,2,2,2,2,2,2,2,2};

    std::cout << "The first 10 even numbers are: ";
    std::partial_sum(v.begin(), v.end(),
                    std::ostream_iterator<int>(std::cout, " "));
    std::cout << '\n';

    std::partial_sum(v.begin(), v.end(), v.begin(), std::multiplies<int>());
    std::cout << "The first 10 powers of 2 are: ";
    for(auto n: v) {
        std::cout << n << " ";
    }
    std::cout << '\n';
}
```

Вывод:

```
The first 10 even numbers are: 2 4 6 8 10 12 14 16 18 20
The first 10 powers of 2 are: 2 4 8 16 32 64 128 256 512 1024
```

См. также

adjacent_difference	вычисляет разницу между соседними элементами в диапазоне (шаблон функции) [править] (https://ru.cppreference.com/mwiki/index.php?title=%D0%A8%D0%B0%D0%B1%D0%BB%D0%BE%D0%BD:cpp/algorithm/dsc_adjacent_difference&action=edit)
----------------------------	---

accumulate	суммирует диапазон элементов (шаблон функции) [править] (https://ru.cppreference.com/mwiki/index.php?title=%D0%A8%D0%B0%D0%B1%D0%BB%D0%BE%D0%BD:cpp/algorithm/dsc_accumulate&action=edit)
-------------------	--

Источник — «https://ru.cppreference.com/mwiki/index.php?title=cpp/algorithm/partial_sum&oldid=32869»