# <numeric>

Defines container template functions that perform algorithms for numerical processing.

#include <numeric>

## Remarks

The algorithms resemble the Standard Template Library (STL) algorithms, but are part of the C++ Standard Library instead. Nevertheless, they are compatible with the STL and, like the STL algorithms, they can operate on a variety of data structures. These include STL container classes—for example, vector and list, and program-defined data structures and arrays of elements that satisfy the requirements of a particular algorithm. The algorithms achieve this level of generality by accessing and traversing the elements of a container indirectly through iterators. The algorithms process iterator ranges that are typically specified by their beginning or ending positions. The ranges referred to must be valid in the sense that all pointers in the ranges must be dereferenceable and within the sequences of each range, and the last position must be reachable from the first by means of incrementation.

The algorithms extend the actions that are supported by the operations and member functions of each of the STL containers and enable interaction with different types of container objects at the same time.

### Functions

|  |  |
| --- | --- |
| accumulate | Computes the sum of all elements in a specified range—including some initial value—by computing successive partial sums, or computes the result of successive partial results that are obtained by using a specified binary operation instead of the sum operation. |
| adjacent\_difference | Computes the successive differences between each element and its predecessor in an input range and outputs the results to a destination range, or computes the result of a generalized procedure where the difference operation is replaced by another specified binary operation. |
| inner\_product | Computes the sum of the element-wise product of two ranges and adds it to a specified initial value, or computes the result of a generalized procedure where the sum and product operations are replaced by other specified binary operations. |
| iota | Stores a starting value, beginning with the first element and filling with successive increments of the value (**value++**) in each of the elements in the interval **[first, last)**. |
| partial\_sum | Computes a series of sums in an input range from the first element through the *i*th element and stores the result of each sum in the *i*th element of a destination range, or computes the result of a generalized procedure where the sum operation is replaced by another specified binary operation. |