Algorithms with Parallelized Versions

A list of all the algorithms which can be parallelized.

Here are the 77 algorithms with parallelized versions.

std::adjacent_difference	std::adjacent_find	std::all_of
std::any_of	std::copy	std::copy_if
std::copy_n	std::count	std::count_if
std::equal	std::exclusive_scan	std::fill
std::fill_n	std::find	std::find_end
std::find_first_of	std::find_if	std::find_if_not
std::for_each	std::for_each_n	std::generate
std::generate_n	std::includes	std::inclusive_scan
std::inner_product	std::inplace_merge	std::is_heap
std::is_heap_until	std::is_partitioned	std::is_sorted
std::is_sorted_until	std::lexicographical_compare	std::max_element
std::merge	std::min_element	std::minmax_element

std::mismatch	std::move	std::none_of
std::nth_element	std::partial_sort	std::partial_sort_copy
std::partition	std::partition_copy	std::reduce
std::remove	std::remove_copy	std::remove_copy_if
std::remove_if	std::replace	std::replace_copy
std::replace_copy_if	std::replace_if	std::reverse
std::reverse_copy	std::rotate	std::rotate_copy
std::search	std::search_n	std::set_difference
std::set_intersection	std::set_symmetric_difference	std::set_union
std::sort	std::stable_partition	std::stable_sort
std::swap_ranges	std::transform	std::transform_exclusive_scan
std::transform_inclusive_scan	std::transform_reduce	std::uninitialized_copy
std::uninitialized_copy_n	std::uninitialized_fill	std::uninitialized_fill_n
std::unique	std::unique_copy	

The 77 algorithms with parallelised versions

> **i Availability of the Parallel STL**

As far as I know, at the time this book is being updated to C++17 (October 2017), there is no standard-conforming implementation of the parallel

STL available yet. Therefore, you have to install third-party frameworks such as HPX. The HPX (High-Performance ParalleX) is a framework that is a general purpose C++ runtime system for parallel and distributed applications of any scale. HPX has already implemented the parallel STL in a different namespace.