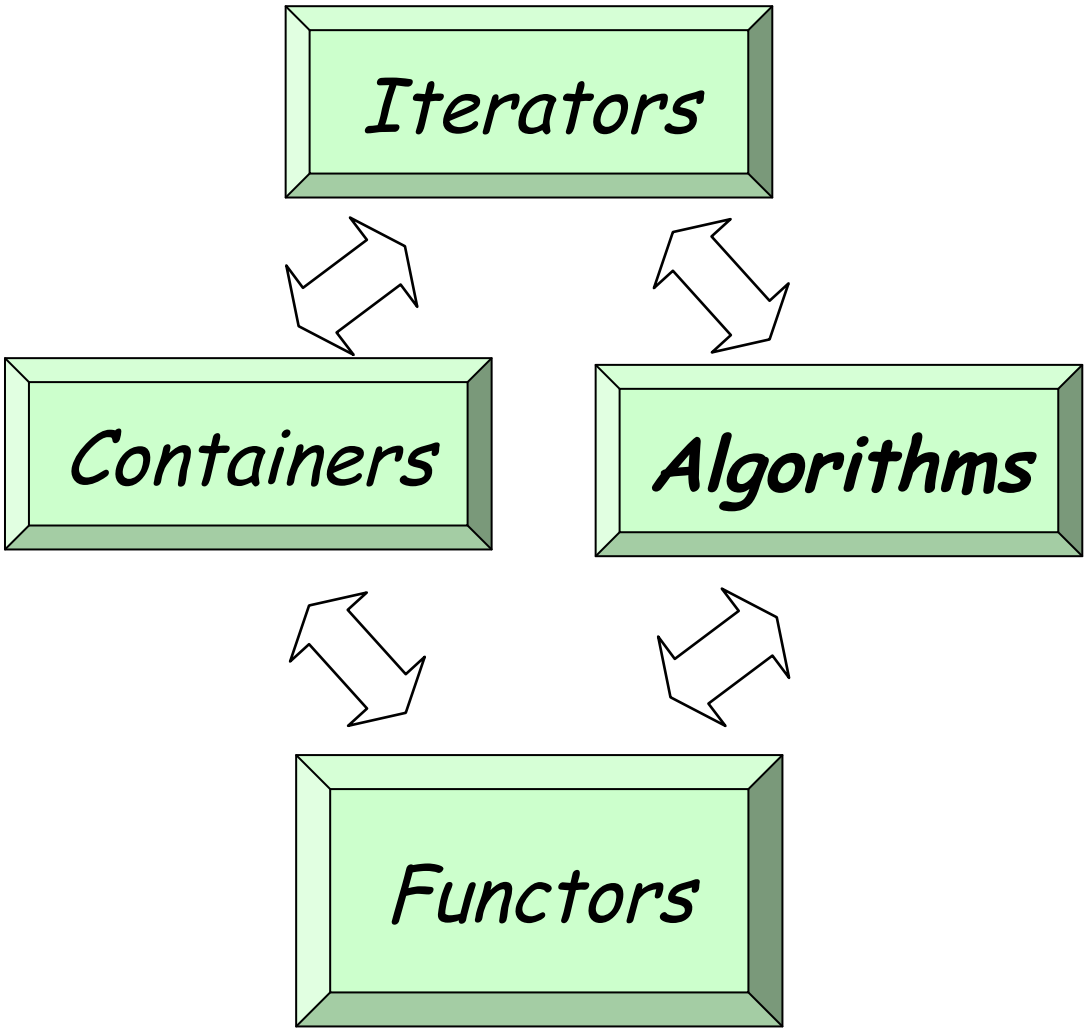
The Standard C++ Library – Algorithms

Based on: Jonathan Boccara, "105 algorithms in less than an hour", CPPCON 2018, https://youtu.be/2olsGf6JIkU

Summarized by: Erel Segal-Halevi

Main Components of STL



Algorithms

- In C++17, there are 105 algorithms that perform common programming tasks.
- We need to know them all.
- *Why?* - otherwise, we will try to implement them ourselves as part of larger projects, and:
  - Make bugs (no time to test..),
  - Produce inefficient code (no time to optimize..)
- *How can we remember them? - use a map →*


Province of Value Queries
count
all_of
any_of
none_of
transform_inclusive_scan
Valley of Partial Sums
transform_exclusive_scan
inclusive_scan
exclusive_scan
partial_sum
Province of Property Queries
Province of 2-Ranges Properties
set_difference
set_symmetric_difference
equal
mismatch
lexicographical compare
is_permutation
Glorious County of Algos on Sets
set_intersection
set_union
includes
merge
for_each
transform
Lonely Islands
uninitialized_value_construct
uninitialized_default_construct
uninitialized_copy
uninitialized_fill
uninitialized_move
destroy
copy_backward
Lands of Queries
find
adjacent_find
transform_reduce
Value Searchers
Lagoon of Folds
reduce
inner_product
equal_range
upper_bound
accumulate
binary_search
lower_bound
sample
adjacent_difference
Peninsula of Raw Memory
Territory of Movers
copy
move
swap_ranges
move_backward
Province of Research
max_element
min_element
Bay of Range Searches
find_end
search
Relative Value Searchers
minmax_element
find_first_of
remove
unique
Island of Structure Changers
fold
fill
generate
replace
Land of Value Modifiers
Secret Runes
stable_*
*_copy
*_n
*_if
is_*_until
is_*
random_shuffle (wreck)
Lands of Permutations
partition_point
partition
reverse
rotate
Shore of Sorting
sort
partial sort
sort_heap
nth_element
inplace_merge
make_heap
pop_heap
push_heap
shuffle
next_permutation
prev_permutation
Province of Heaps
The World of C++ STL Algorithms
Fluent {C++}
Fluent {C++}
fluentcpp.com
https://youtu.be/2olsGf6JIkU?t=475

Queries (10)

Algorithms that calculate a value related to the collection, without modifying it.

Set Algorithms (11)

Operations on any sorted collection (including std::set but also sorted vectors)

Permutations (12)

Algorithms that move elements around the collection, without changing their values:

Movers (13)

Algorithms that copy/move things between collections.

Value modifiers (14)

Modify values in a collection.

Runes (15)

Words you can add to an algorithm to get a new algorithm:

Structure changers (16)

STL algorithms work on iterators, so they cannot change the size!

Solution: move the *remaining* elements to the *beginning* of the collection, and return an iterator to the *new end*.

Others (17)

What now?

- Use algorithms instead of loops.
- Read the documentation: preconditions, postconditions, complexity
- Invent your own algorithms!