Data Structures with C++ STL

Some Prerequisites for this Session

- Basic knowledge of C++
- Object Oriented Techniques
- An idea about basic data structures (their basic properties and stuff, just an idea would do).

Uses of Templates

- Generic Functions
- Generic Classes and Structures
- Generic Type Defines

What does C++ STL contain?

- Containers
- Iterators
- Algorithms

* All of these can be found in one header, <bits/stdc++.h>

Note: It is a GNU implementation for the GNU G++ compiler for UNIX based systems. It is not a standard C++ implementation.

Some Standard Utilities

- pair<A, B>
- tuple<A, B, C, ...>

Iterators

- They are basically wrappers for pointers in a container.
- They can possess the property of “Random Access”, “Bidirection”, ... etc.

Containers

- They are linear (contiguous) or non – linear structures used to store a particular class of elements (data).
- Some properties (methods) are common to all of them.

Dynamic Arrays

- Vectors
- Stack
- Queue
- Deque

Online and Sorted Containers

- Set
- Map
- Multiset
- Multimap

The last two are more or less the same with an added feature to support element duplicacy.

Online and Unsorted Containers

- Unordered Set
- Unordered Map
- Unordered Multiset
- Unordered Multimap

Custom Comparators and Hashers

- We can define our own rules for the structure of a particular container, or for directing a particular search.

A special online Container

Priority Queue

Code shown during Session

Generic Functions: https://ideone.com/GNjA3K

Generic Class and Type: https://ideone.com/OeNrXR

Tuples: https://ideone.com/CRfcvB

Dynamic Arrays: https://ideone.com/qJ8bFO

All testing done here: https://ideone.com/037Nuo

Online Sorted Data Structures: https://ideone.com/reZElb

Note: The last one could not be shown during the session.

Compilation

This is how I compiled code during the session:

g++ -no-pie -fno-pie -std=gnu++17 -O3 -Wall -D LOCAL -g -o executable_name source_name.cpp

Resource

http://www.cplusplus.com/

Thank You!