② Introduction to C++

→ Compiled programming language
→ C++ 11 turns C++ into a totally modern programming language.

→ C++ used in embedded devices, IOT, automotive sectors

→ C++ 17 was a major release but C++ 11 was one of major release.

→ C++ 11 includes :-
- *) Move Sematics
- *) Variadic templates
- *) Initialiser lists
- *) auto keyword
- *) Lambda expression
- *) null pointer
- *) constant expression
- *) range-based for loops
- *) smart pointers.

→ C++ 17 includes :-
- *) Standard filesystem with STL
- *) Standard string View
- *) Parallel implementation of many STL
- *) Inline variables

Subject: / /

→ C++ 20 : Minor release that includes

*) Generic programming with templates

Standard Library

→ C++ standard library is a collection of classes and functions, which are written in the core language and part of the C++ ISO standard itself.

→ It is preferable to utilize functionality that already exists in the standard library, instead of implementing it from scratch.

→ C++ core guidelines (SL.1) states that :-

"Use libraries wherever possible"
Reason : Save time. Don't reinvent the wheel. Don't replicate the work of others.

and SL.2

"Prefer the standard library to other libraries"
Reason : More people know the standard library. It is more likely to be stable, well maintained, and widely available than your own code or most other libraries.

Subject: / /

Namespace

→ STL functions and classes exist in the std:: namespace.

Eg: std::vector

→ To use a STL feature, the corresponding header file should be included

Eg: #include <vector>

```cpp
#include <iostream>
#include <vector>


int main() {
    std::vector<int> intNum = {15, 5, 8};
    return 0;
}

```

Compilation

→ C++ is a compiled programming language, which means that programmers use a program to compile their human-readable source code into machine-readable object and executable files.

→ The program that performs this task is called compiler.

Subject: / /

→ C++ doesnot have any official compiler. Instead, there are many different compilers.

GNU compiler collection

→ It is a popular, open-source, cross platform compiler from the larger GNU project.

→ In particular, g++ is a command line executable that compiles C++ source code and automatically links the C++ standard library.

Linking

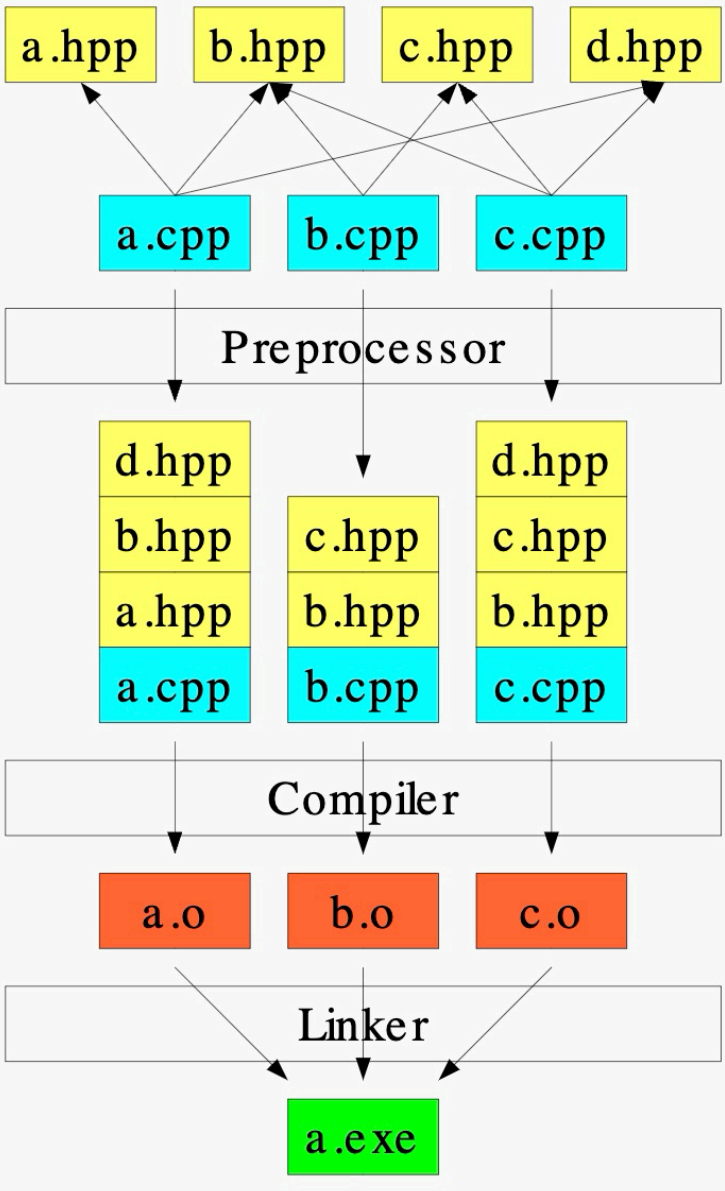
→ In order to use classes and functions from the C++ STL, the compiler must have access to a compiled version of the standard library, stored in object files.

→ Most compilers including gcc (GNU compiler collection) include those object files as part of the installation process.

→ In order to use the STL facilities, the compiler must link the standard library object files to the object files created from the programmer's source code.

Subject: / /

→ then once the linking is complete, it is able to generate a standalone executable.



C++ Compilation Process (Wikimedia)