C++

HS20201

Marco Agostini

Computer Science
University of Applied Sciences of Eastern Switzerland
September 2021

Contents

1 Introduction

1.1 Why C++?

- Work al almost all platforms from a micro controller to the main frame
- Multi-paradigm language with zero-cost abstraction
- High-level abstraction facilities
- The concepts from C++ can mostly be applied to any other programming language

1.2 Undefined Behaviour

The undefined behaviour is defined in the C++ standard (funny, isn't it?). C++ has no garbage collector. If in C++ something is written wrong and the compiler doesn't detect it: undefined behaviour can occur.

1.3 C++ Compilation Process

***.cpp files for source code**

- Also called "Implementation File"
- Function implementations (can be in .h files as well)
- Source of compilation - aka "Translation Unit"

***.h files for interfaces and templates**

- Called "Header File"
- Declarations and definitions to be used in other implementation files.
- Textual inclusion through a pre-processor (C++20 will incorporate a "Module" mechanism)
- #include "header.h"

3 Phases of Compilation

- **Preprocessor** – Textual replacement of preprocessor directives (#include)
- **Compiler** – Translation of C++ code into machine code (source file to object file)
- **Linker** - Combination of object files and libraries into libraries and executables

1.4 Declarations and Definitions

All things with a name that you use in a C++ program must be declared before you can do so!
Declaring Functions
$< return - type >< function - name > (< parameters >);$