B.3 — Introduction to C++17

BY ALEX ON MARCH 31ST, 2018 | LAST MODIFIED BY ALEX ON MAY 6TH, 2018

What is C++17?

In September of 2017, the **ISO (International Organization for Standardization)** approved a new version of C++, called C++17. C++17 contains a fair amount of new content

New improvements in C++17

For your interest, here's a list of the major improvements that C++17 adds. Note that this list is not comprehensive, but rather intended to highlight some of the key improvements of interest.

- __has_include preprocessor identifier to check if optional header files are available (no tutorial yet)
- if statements that resolve at compile time (no tutorial yet)
- Initializers in if statements and switch statements (no tutorial yet)
- inline variables (no tutorial yet)
- Fold expressions (no tutorial yet)
- Nested namespaces can now be defined as namespace X::Y (**4.3b -- Namespaces**)
- Removal of std::auto_ptr and some other deprecated types
- static_assert no longer requires a diagnostic text message parameter (**7.12a -- Assert and static_assert**)
- std::any (no tutorial yet)
- std::byte (no tutorial yet)
- std::filesystem (no tutorial yet)
- std::optional (no tutorial yet)
- std::shared_ptr can now manage C-style arrays (but std::make_shared can't create them yet) (**15.6 -- std::shared_ptr**)
- Structured binding declarations (**7.4a -- Returning values by value, reference, and address**, but could use a full lesson)
- Template deduction for constructors (no tutorial yet)
- Trigraphs have been removed (no tutorial yet)
- typename can now be used (instead of class) in a template template parameter (no tutorial yet)
- UTF-8 (u8) character literals (no tutorial yet)

Appendix C -- The end?

Index

B.2 -- Introduction to C++14

C++ TUTORIAL | PRINT THIS POST

2 comments to B.3 — Introduction to C++17

nascardriver
May 1, 2018 at 6:20 am · Reply