

Agenda "Modern C++" / tisoware Reutlingen / 24.-25.10.2024

Donnerstag

Move-Semantik

auto

Lambdas

Initialisierung (Brace)

Initialisierungs-Listen (std::initializer_list)

Algorithmen

Range-Based for-Loop

Smart-Pointer

RVO-Optimierung

constexpr

Utility-Klassen (std::optional, std::variant, std::any, std::array)

Type-Traits

Generische Lambdas

Freitag

Catch2 (Unit Test)

RAII

Multi-Threading

Kurze Einführung "Variadic Templates"

Wozu: `emplace` // `std::make_unique`

Kurze Einführung "Folding"

Wozu: Performanz

Utility-Klassen (std::string_view vs. std::string)

Exception Guarantee

Copy-Swap Idiom

Allgemeines zu Performanz

const

emplace

reserve bei std::vector

RVO

virtual vs. non-virtual

std::move

Soweit möglich Referenzen verwenden (auto&)

Die Wahl des richtigen Containers

STL-Algorithmen einsetzen (std::fill, std::copy, ...)

std::initializer_list versus std::vector

noexcept

Ausblick

std::println

Module