

# Zaawansowane techniki programowania w języku C++



## Odbiorcy szkolenia

Szkolenie skierowane jest do programistów mających więcej niż 3 lata doświadczenia w programowaniu C++



## Korzyści

Umiejętność samodzielnego tworzenia bardziej złożonych programów w C++ oraz poznanie najnowszych standardów



## Program szkolenia

1. C++11 Features:
  - move semantic
  - lambda
  - static\_assert
  - Uniform initialization
  - In-class initialization of non-static variables
  - initializer\_list
  - alias
  - Template alias
  - C'tor inheritance
  - Attributes
  - decltype
  - auto deduction

2. - C++14 Features:
  - decltype(auto)
  - Variable templates
  - Binary literals (Finally!)
  - Digit separators
3. - C++17 Features:
  - Nested namespace definitions
  - Class template argument deduction
  - Selection statements with initializer
  - Unified initialization
  - Structural binding
  - Folding
  - constexpr
  - Filesystem
4. C++20 Features:
  - ranges
  - modules
  - concepts
  - operator<=>,
  - designated initializers,
  - attributes
  - pack-expansion in lambdas (how to avoid copy)
  - template syntax for lambdas
  - uniform erasure
  - How to log useful informations in fast way using C++20 (source\_location), substitution for old macros.
  - bit operations
5. Smart pointers:
  - unique\_ptr
  - shared\_ptr
  - weak\_ptr
  - Implementation details
  - good practise
6. Templates:
  - Write generic code
  - Deduction guidelines



Oczekiwane przygotowanie uczestnika

- Znajomość języka C++ na poziomie średnio-zaawansowanym (iteratory, kontenery, algorytmy, podstawy szablonów itd...)
- Obsługę kompilatora (clang/ g++)
- Podstawowa znajomość linuxa (jak przechodzić pomiędzy katalogami, tworzenie pliku itd...)
- Umiejętność pobrania repozytorium z githuba
- Korzystanie z CMake



## Czas trwania

3 dni / 14 godzin

## Język

- Szkolenie: polski