

## C++ course (2020) outline

By Oleksiy Grechnyev, IT-JIM

Course level: Basic to intermediate, knowledge of any other programming language helps

1. Introduction  
Why C++? Compilers vs interpreters. Hello World and other examples. C++ compilers (gcc, clang/llvm, CL) and IDEs. Boost. C++ bibliography.
2. C++ Language Basics 1  
C++ versions (98, 03, 11, 14, 17). Built-in types. Operators. Statements. CMake example. using/typedef. auto
3. C++ Language Basics 2  
References. Pointers. const and constexpr modifier. Functions. Overloading. Headers. Namespaces.
4. Classes 1  
Access modifiers : *public*, *private*, *protected*. Constructors. Destructors. Friends. Inheritance. Overriding, virtual and abstract methods. Multiple inheritance.
5. Smart Pointers + Miscellanea 1  
Object life cycle. *unique\_ptr*. *shared\_ptr*. *weak\_ptr*. Exceptions, try+catch. pair, any, optional. enum (class). Type casts.
6. Containers + Miscellanea 2  
array, vector, string, string\_view, map. Iterators. Algorithms. Date + time. Random numbers.
7. Lambda expressions. IO streams.  
std::function. Function pointers. Functors. Lambda expressions. IO streams.
8. Classes 2  
How to write your own class. Operator overloading. Copy + move constructors.
9. Using cmake and make  
CMake features. Static and dynamic libraries. Using make. Various nice C++ libraries.
10. Concurrency. Templates. Move semantics.  
thread, async, future, atomic, mutex. Templates. move, swap, rvalue references.