Advanced C++ programming:

First thing first

0.1 Different versions of C++ standard

Because of the growing complexity of computer systems, there’s a need to develop new versions of programming languages, including C++.

On the one hand, C++ is not an old language; on the other hand, from the perspective of some of ultramodern languages, it may be considered very outdated. Still, there have been long periods of C++ stability, and nowadays we can say that C++ is a highly developed language.

Let’s take a general look at how C++ has evolved over the years, and at the main features different C++ standard versions have introduced.

0.2. This course and different versions of C++ std

Throughout this course we’re going to use some of the modern features that we showed you on the previous slide. Whenever we find it useful, we’ll let you know about this .

All the code that we’re going to use in our examples has been compiled with a C++14 standard compliant compiler, and it may happen that the code won’t work with some older compilers (well, if you have a very rusty one, then maybe it’s time to recognize the signs of the future and change to a more contemporary compiler?).

Remember: some of the new features will significantly improve, and shorten, your programs.