



**“AZƏRBAYCAN HAVA YOLLARI” CJSC NATIONAL AVIATION ACADEMY**

**Individual Work № 3:**

**Topic: C / C ++ compilers**

**Subject: System software and Operating systems-2**

**Teacher: Mehemmed Shahmaliyev**

**Group: 1459i Student: Aliyeva Ilaha**

**Date: Signature**

**Baku 2022**

**Introduction to Best C++ Compilers**

This article explains many features with respect to the compilers. Here we explain how compilers work with c++ language. Why do we need compilers? Here we go with the answers; As we all know the computer understands a machine language (set of instructions) with the combination of zero’s and one’s. What we write as source code is a high-level language. We need some translator or interpreter to make understand the computer. The easiest way to interact with it is by the compiler. Each c++ file is compiled into an object file. The initial step done by the compiler is to run the pre-processor followed by the compilation phase and gives out the object file. C++ code has thousands of lines with a large number of template programming. Therefore, it is necessary for the user to achieve higher productivity.

**Top 8 C++ Compilers**

C++ is named as a compiled language due to a lot of work done by the compilers. here is the list of c++ compilers which will do for both c and c++, differed by their file extension (for c it is (.C) extension, For C++ it is .cpp extension). Using the editor’s C++ source file is generated and the compiler is invoked to give an executable file and finally, the name of the file is executed to produce the result. Below is a list of different compilers allowing us to compile the source code on the different platforms:

Let’s look at the top 7 best compilers in 2019.

* MinGW / GCC
* Borland c++
* Dev C++
* Embracadero
* Clang
* Visual C++
* Intel C++
* Code Block

We have briefly described all the one in the following list. Below is a list of different compilers.

#### 1. MinGW

It’s an opensource tool with no third-party requirements and works well with the development of Microsoft windows. It has GCC compilers to include C, C++, Fortran language compilers. Among many other tools, this compiler is liked most by the user due to the high level of portability available in GCC by ANSI Compliance. Windows 32 or min32 makes to create an individual project and added with different packages and licensed in their own versions. G++ provides us with the complete source code. A key benefit of it is fast and simple and requires DLL libraries.

#### 2. Borland C++

This is the oldest and cheaper compiler which do not have any competitors in the market. Borland is a C++ integrated development environment and it is most widely used in the 90’s works well in [MS-DOS Prompt](https://www.educba.com/what-is-dos/) and Windows. They make simple editing features making their code easier and simple.

#### 3. Dev C++

Dev C++ takes a combination of GCC along with Cygwin port as a compiler helps in editing and compiling resource files. It’s an open-source environment and sufficient for the new learners of the C++ program. This IDE or compiler is written in [Delphi and comes with DevOps](https://www.educba.com/what-is-devops/). The key benefit of it is can be installed and execute on a flash drive makes a full-featured Compiler.

#### 4. Embarcadero C++ builder

It’s an open-source Compiler for windows meant for commercial purposes. It supports language extensions, libraries, good UI design and considered to be a cross-platform from Windows, IOS, Android. Students can utilize it at free of charge as they have published with a new command-line compiler. This tool is enabling to deliver the applications faster and insecure platforms in the competitive market.

#### 5. Clang

Clang compiler is preferred to be easily understandable compiler provided with front-end with more fast and reduced memory adapted with a BSD license. A good feature of Clang is its GCC compatibility and its design is based on LLVM. It is built for a better analysis of the code with faster compilation. The three phases of a compiler include a parser to build the Abstract Syntax tree at the front end, the second phase includes the optimizer, the final is the back end. Therefore, they are considered to be the important mainstream compiler in the industry. Clang is considered to be a good compiler to build a tool and to reuse, integrate with other projects too.

#### 6. Visual C++

The development of Visual C++ has migrated to new technology by Microsoft in the year 1993. Microsoft strategy has achieved new features with the class library which gains the greatest productivity and gives an integrated environment for professional C++ developers. A key feature of Visual C++ is the development of MFC architecture which provides the fastest executables, developing windows-based applications. Most importantly, to get better compatibility MFC 2.0 is been used for a good subset of C++ to make the framework much portable. They give you a family of products and provides a path to develop windows applications in 32-bit to have the safest run on multiple hardware and software, meanwhile incorporating OLE development where users burden is reduced on writing code. And it can be executable on commercially provided compiler supporting component-based software to evolve.

#### 7. Intel C++

Intel C++ or named as ICC is developed by Intel corporation with embedded New Intel architectures, this program compiles C and C++ and comes with a commercial license, the version 11.0 of it provides RPMs. They contain highly targeted Cache work and SIMD instructions. This compiler processes the job faster and more effective and advantageous than GCC. Intel C++ has an identical performance where other compiler faces scalar instructions. The key attribute of Intel is their flexibility, switching between the compilers is easy and preferred much by the software developers.

#### 8. Code block

The code block is a free cross-platform c++ compiler and supports multiple platform compilers. It includes features like debugger, breakpoints in the code and multi-compilers. A super benefit of the Code block is they operate on different Operating Systems and do not require translating languages for their operations.

### Conclusion

To conclude, in this article we have seen how the compiler does the job with c++. This will give us more insights into the c++ development process. Delegates interested in learning c, [c++ programming languages](https://www.educba.com/what-is-a-programming-language/) make use of this free compilers to build their codes. A good compiler is chose based on two aspects they are compilation speed and the time taken for compilation for huge projects. The compiler should be efficient in performance and work harder to give out of code.