COMPUTER LANGUAGES

S. Rudenko – Sumy State University, group IP – 12
S.V. Mikhno – ELA

The basic knowledge of programming is not possible without the knowledge of its history.

The first programming languages predate the modern computer. From the first, the languages were codes. Herman Hollerith realized that he could encode information on punch cards when he observed that railroad train conductors would encode the appearance of the ticket holders on the train tickets using the position of punched holes on the tickets.

The first computer codes were specialized for the applications. In the first decades of the twentieth century, numerical calculations were based on decimal numbers.

In the 1940s the first recognizably modern, electrically powered computers were created. The limited speed and memory capacity forced programmers to write hand tuned assembly language programs. It was soon discovered that programming in assembly language required a great deal of intellectual effort and was error-prone.

Programs written in high-level languages have a number of advantages. First, they are easier to learn, understand, and modify than machine and assembly language programs. Second, high-level languages do not depend on the machine language of any one type of computer.

For example, when programming is a driver for an operating system, obviously we will use low-level programming. Whereas when programming great applications usually a higher level it is used, a combination of critic parts written in low level languages and others in higher.

Computers can run only programs written in the language that they understand their individual machine language.

In order to support a higher-level language, a special computer program must be developed that translates the statements of the program developed in the higher-level language into a form that the computer can understand – in other words, into the particular instructions of the computer.

The approach or method that is used to solve the problem is known as an algorithm. For example, if we wish to develop a program that tests if a number is odd or even, then the set of statements which solves the problem becomes the program. The method that is used to test if the number is even or odd is the algorithm.

To develop a program, to solve a particular problem, we first express the solution to the problem in terms of an algorithm and then develop a program, which implements that algorithm. Then we can proceed to write the instructions necessary to implement the algorithm on a particular computer system.

C++ is one of the most popular programming languages with application domains including systems software, application software, device drivers, embedded software, high-performance server and client applications, and entertainment software such as video games.

Several groups provide both free and proprietary C++ compiler software, including the GNU Project, Microsoft, Intel and Embarcadero Technologies. C++ has greatly influenced many other popular programming languages, most notably C# and Java.

C++ is also used for hardware design, where the design is initially described in C++, then analyzed, architecturally constrained, and scheduled to create a register-transfer level hardware description language via high-level synthesis.