

## Original

Unfortunately for us, computers can't understand spoken English or any other natural language. **The only language they can understand directly is machine code**, which consists of 1s and 0s (binary code).

**Machine code is too difficult to write.** For this reason, **we use symbolic languages to communicate instructions to the computer.** For example, **assembly languages** use abbreviations such as ADD, SUB, MPY to represent instructions. The **program is then translated into machine code by a piece of software called an assembler.** **Machine code and assembly languages are called low-level languages** because they are closer to the hardware. They are quite complex and restricted to particular machines. **To make the programs easier to write,** and to overcome the problem of intercommunication between different types of computer, **software developers designed high-level languages, which are closer to the English language.** Here are some examples:

- **FORTRAN** was developed by IBM in 1954 and **is still used for scientific and engineering applications.**
- **COBOL** (Common Business Oriented Language) was developed in 1959 and **is mainly used for business applications.**
- **BASIC** was developed in the 1960s **and was widely used in microcomputer programming because it was easy to learn.** Visual BASIC is a modern version of the old BASIC language, used to build graphical elements such as buttons and windows in Windows programs.
- **PASCAL** was created in 1971. **It is used in universities to teach the fundamentals of programming.**
- **C** was developed in the 1980s at AT&T. **It is used to write system software, graphics and commercial applications.** C++ is a version of C which incorporates object-oriented programming: the programmer concentrates on particular things (a piece of text, a graphic or a table, etc. ) and gives each object functions which can be altered without changing the entire program. For example, to add a new graphics format, the programmer needs to rework just the graphics object. This makes programs easier to modify.
- **Java** was designed by Sun in 1995 to run on the Web. Java applets provide animation and interactive features on web pages.

**Programs written in high-level languages must be translated into machine code by a compiler or an interpreter.** A **compiler** translates the source code into object code — that is, **it converts the entire program into machine code in one go.** On the other hand, an **interpreter translates the source code line by line as the program is running.**

**It is important not to confuse programming languages with markup languages, used to create web documents.** Markup languages use instructions, known as markup tags, to format and link text files. Some examples include:

- **HTML**, which allows us to describe how information will be displayed on web pages.
- **XML**, which stands for EXtensible Markup Language. While HTML uses pre-defined tags, **XML enables us to define our own tags;** it is not limited by a fixed set of tags.
- **VoiceXML**, which makes Web content accessible via voice and phone. VoiceXML is used to create voice applications that run on the phone, whereas HTML is used to create visual applications (for example, web pages).

## Translation

К сожалению для нас, компьютеры не понимают английский или любой другой естественный язык. Единственный язык, который напрямую понимают компьютеры — это машинный язык, состоящий из нулей и единиц (двоичный код).

Машинный код труден для написания, поэтому мы используем символьные языки, чтобы сообщить компьютеру, какие инструкции он должен выполнить. Например, языки ассемблера для представления инструкций используют аббревиатуры, такие как ADD, SUB, MPY. Затем программы на языке ассемблера преобразуются в машинный код специальной программой, которая называется ассемблер. Машинный код и языки ассемблера называются низкоуровневыми языками, потому что они очень близки к «железу». Они довольно сложные и работают только на определенных машинах. Чтобы сделать программы проще для написания и решить проблему переноса программ между разными компьютерами, программисты разработали высокоуровневые языки программирования, которые ближе к английскому языку. Вот несколько примеров:

- FORTRAN был разработан IBM в 1954 году и до сих пор используется для научных и инженерных целей.
- COBOL (общий бизнес-ориентированный язык) был разработан в 1959 году и в основном используется для бизнес-приложений.
- BASIC был разработан в 1960-х годах и широко использовался в программировании микрокомпьютеров, потому что его было легко освоить. Visual BASIC — это современная версия старого языка BASIC, используемая для построения графических элементов, таких как кнопки и окна в программах Windows.
- Паскаль был создан в 1971 году. Он используется в университетах преподавать основы программирования.
- C был разработан в 1980-х годах в AT&T. Он используется для написания системного программного обеспечения, графики и коммерческих приложений. C++ — это версия языка C, которая включает в себя объектно-ориентированное программирование: программист концентрируется на определенных вещах (фрагменте текста, графике или таблице и т. д.) и дает каждому объекту функции, которые могут быть изменены без изменения всей программы. Например, чтобы добавить новый графический формат, программисту нужно переработать только графический объект. Это облегчает модификацию программ.
- Java была разработана компанией Sun в 1995 году для работы в Интернете. Java-апплеты обеспечивают анимацию и интерактивные функции на веб-страницах.

Программы, написанные на языках высокого уровня, должны быть переведены в машинный код компилятором или интерпретатором. Компилятор переводит исходный код в объектный код — то есть он преобразует всю программу в машинный код за один раз. С другой стороны, интерпретатор переводит исходный код строка за строкой по мере выполнения программы.

Важно не путать языки программирования с языками разметки, используемыми для создания веб-документов. Языки разметки используют инструкции, известные как теги разметки, для форматирования и связывания текстовых файлов. Некоторые примеры включают в себя:

- HTML, который позволяет нам описать, как информация будет отображаться на веб-страницах.
- XML, что означает расширяемый язык разметки. В то время как HTML использует предопределенные теги, XML позволяет нам определять наши собственные теги; он не ограничен фиксированным набором тегов.
- VoiceXML, который делает веб-контент доступным через голос и телефон. VoiceXML используется для создания голосовых приложений, работающих на телефоне, в то время как HTML используется для создания визуальных приложений (например, веб-страниц).