

How computer understand user

What is a compiler?

A compiler is a special program that translates a programming language's [source code](#) into machine code, bytecode or another programming language. The source code is typically written in a high-level, human-readable language such as [Java](#) or [C++](#). A programmer writes the source code in a code editor or an integrated development environment ([IDE](#)) that includes an editor, saving the source code to one or more text files. A compiler that supports the source programming language reads the files, analyzes the code, and translates it into a format suitable for the target platform.

Compilers that translate source code to [machine code](#) target specific operating systems and computer architectures. This type of output is sometimes referred to as object code (which is not related to [object-oriented programming](#)). The outputted machine code is made up entirely of binary bits -- 1s and 0s -- so it can be read and executed by the processors on the target computers. For example, a compiler might output machine code for the Linux x64 platform or Linux ARM 64-bit platform

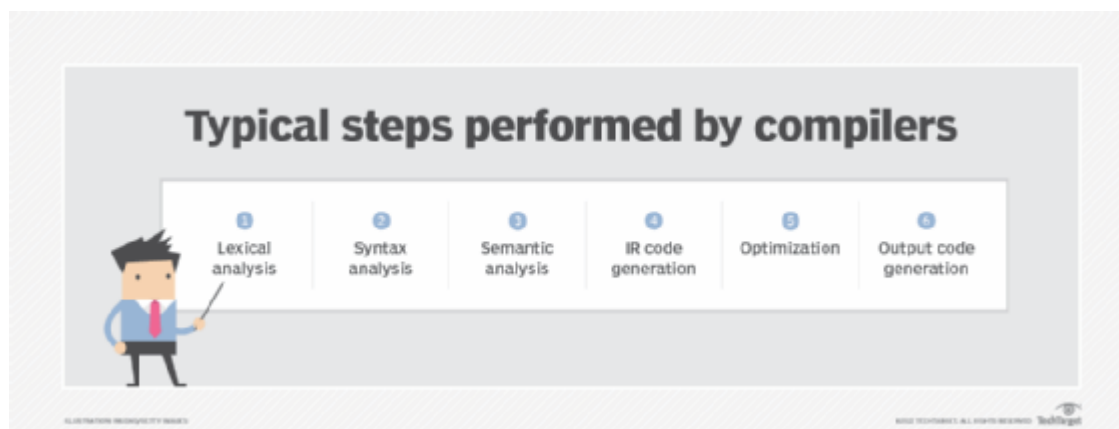
How does a compiler work?

Compilers vary in the methods they use for analyzing and converting source code to output code. Despite their differences, they typically carry out the following steps:

- **Lexical analysis.** The compiler splits the source code into lexemes, which are individual code fragments that represent specific patterns in the code. The lexemes are then tokenized in preparation for [syntax](#) and semantic analyses.

How computer understand user

- **Syntax analysis.** The compiler verifies that the code's syntax is correct, based on the rules for the source language. This process is also referred to as parsing. During this step, the compiler typically creates abstract syntax trees that represent the logical structures of specific code elements.
- **Semantic analysis.** The compiler verifies the validity of the code's logic. This step goes beyond syntax analysis by validating the code's accuracy. For example, the semantic analysis might check whether variables have been assigned the right types or have been properly declared.
- **IR code generation.** After the code passes through all three analysis phases, the compiler generates an intermediate representation (IR) of the source code. The IR code makes it easier to translate the source code into a different format. However, it must accurately represent the source code in every respect, without omitting any functionality.
- **Optimization.** The compiler optimizes the IR code in preparation for the final code generation. The type and extent of optimization depends on the compiler. Some compilers let users configure the degree of optimization.
- **Output code generation.** The compiler generates the final output code, using the optimized IR code.



How computer understand user

What Does Interpreter Mean?

An interpreter is a computer program that is used to directly execute program instructions written using one of the many high-level programming languages.

The interpreter transforms the high-level program into an intermediate language that it then executes, or it could parse the high-level source code and then performs the commands directly,

Difference between compiler and interpreter

Interpreter translates just one statement of the program at a time into machine code.

Compiler scans the entire program and translates the whole of it into machine code at once.

An interpreter takes very less time to analyze the source code. However, the overall time to execute the process is much slower.

A compiler takes a lot of time to analyze the source code. However, the overall time taken to execute the process is much faster.

An interpreter does not generate an intermediary code. Hence, an interpreter is highly efficient in terms of its memory.

A compiler always generates an intermediary object code. It will need further linking. Hence more memory is needed.

Keeps translating the program continuously till the first error is confronted. If any error is spotted, it

A compiler generates the error message only after it scans the complete program and hence

How computer understand user

stops working and hence debugging becomes easy.

debugging is relatively harder while working with a compiler.

Interpreters are used by programming languages like Ruby and Python for example.

Compilers are used by programming languages like C and C++ for example.

What is Programming?

A simple answer would be, “Programming is the act of instructing computers to carry out tasks.” It is often referred to as **coding**.

So then, what is a **computer program**? A computer program is a sequence of instructions that the computer executes.

Computer in the definition above is any device that is capable of processing code. This could be smartphones, ATMs, the Raspberry Pi, Servers to name a few.

The Natural Language of the Computer

Machines have their natural language like humans do. Computers do not understand the human language. The natural language of computers is the binary code — 1 and 0. These represent two states: **on (1)** and **off (0)**.

How computer understand user