1. A translator is a programming language processor that converts a computer program from one language to another. It takes a program written in source code and converts it into machine code. It discovers and identifies the error during translation.

Purpose of Translator

It translates high-level language program into a machine language program that the central processing unit (CPU) can understand. It also detects errors in the program.

Different Types of Translators

There are 3 different types of translators as follows:

Compiler

A compiler is a translator used to convert high-level programming language to low-level programming language. It converts the whole program in one session and reports errors detected after the conversion. Compiler takes time to do its work as it translates high-level code to lower-level code all at once and then saves it to memory.

A compiler is processor-dependent and platform-dependent. But it has been addressed by a special compiler, a cross-compiler and a source-to-source compiler. Before choosing a compiler, user has to identify first the Instruction Set Architecture (ISA), the operating system (OS) and the programming language that will be used to ensure that it will be compatible.

Interpreter

Just like a compiler, is a translator used to convert high-level programming language to low-level programming language. It converts the program one at a time and reports errors detected at once, while doing the conversion. With this, it is easier to detect errors than in a compiler. An interpreter is faster than a compiler as it immediately executes the code upon reading the code.

It is often used as a debugging tool for software development as it can execute a single line of code at a time. An interpreter is also more portable than a compiler as it is not processor-dependent, you can work between hardware architectures.

Assembler

An assembler is is a translator used to translate assembly language to machine language. It is like a compiler for the assembly language but interactive like an interpreter. Assembly language is difficult to understand as it is a low-level programming language. An assembler translates a low-level language, an assembly language to an even lower-level language, which is the machine code. The machine code can be directly understood by the CPU.

|  |  |  |
| --- | --- | --- |
| Assembler | Compiler | Interpreter |
| Software that converts programs written in assembly language into machine language | Software that converts programs written in high-level language into machine language | Software that translates a high-level language program into machine language |
| Convert assembly language program to machine language | Convert the whole high-level language program to machine language at a time | Convert the high-level language program to machine language line by line |
| Used by assembly language | Used by C, C++ | Used by Ruby, Perl, python ,PHP |

The main advantage of high-level languages over low-level languages is that they are easier to read, write, and maintain. Ultimately, programs written in a high-level language must be translated into machine language by a compiler or interpreter. The first high-level programming languages were designed in the 1950s.