**Interpreter vs Compiler**

[**https://www.geeksforgeeks.org/difference-between-compiler-and-interpreter/**](https://www.geeksforgeeks.org/difference-between-compiler-and-interpreter/)

|  |  |
| --- | --- |
| Interpreter | Compiler |
| Translates program one statement at a time. | Scans the entire program and translates it as a whole into machine code. |
| The overall execution time is comparatively slower than compilers. | The overall execution time is comparatively faster than interpreters. |
| No Object Code is generated, hence are memory efficient. | Generates Object Code which further requires linking, hence requires more memory. |
| As it scans the code in one go, the errors (if any) are shown at the end together. | Considering it scans code one line at a time, errors are shown line by line. |
| The machine code is stored in the disk storage. | Machine code is nowhere stored. |

Source Program->Compiler->Object Program

Source Program->Intermediate Code->Interpreter

**Machine language** is the low level programming language. Machine language can only be represented by 0s and 1s. In earlier when we have to create a picture or show data on the screen of the computer then it is very difficult to draw using only binary digits(0s and 1s). For example: To write 120 in the computer system its representation is 1111000. So it is very difficult to learn. To overcome this problem the assembly language is invented.

**Assembly language**is the more than low level and less than high-level language so it is intermediary language. Assembly languages use numbers, symbols, and abbreviations instead of 0s and 1s.For example: For addition, subtraction and multiplications it uses symbols likes Add, sub and Mul, etc.

| Machine Language | Assembly Language |
| --- | --- |
| Machine language is only understood by the computers. | Assembly language is only understand by human beings not by the computers. |
| In machine language data only represented with the help of binary format(0s and 1s), hexadecimal and octadecimal. | In assembly language data can be represented with the help of mnemonics such as Mov, Add, Sub, End etc. |
| Machine language is very difficult to understand by the human beings. | Assembly language is easy to understand by the human being as compare to machine language. |
| Modifications and error fixing cannot be done in machine language. | Modifications and error fixing can be done in assembly language. |
| Machine language is very difficult to memorize so it is not possible to learn the machine language. | Easy to memorize the assembly language because some alphabets and mnemonics are used. |
| Execution is fast in machine language because all data is already present in binary format. | Execution is slow as compared to machine language. |
| There is no need of translator.The machine understandable form is the machine language. | Assembler is used as translator to convert mnemonics into machine understandable form. |

<https://www.geeksforgeeks.org/difference-between-machine-language-and-assembly-language/#:~:text=Assembly%20languages%20use%20numbers%2C%20symbols,%2C%20sub%20and%20Mul%2C%20etc.&text=Machine%20language%20is%20only%20understand,beings%20not%20by%20the%20computers>.

(Differences Link)