Opeyemi Nasir Firdaus
17/sci01/051
CSC 310

Firstly,
An assembly language consists of a set of symbols and letters and requires translation to machine language. Both machine code and assembly languages are hardware specific while A high-level language is a programming language that uses English and mathematical symbols in its instructions.
Secondly,
Both machine code and assembly languages are hardware specific while a high-level language is a programming language that uses English and

mathematical symbols in its instructions. To execute a program in a high-level language, it can be compiled or interpreted.

Below is a table that also shows the major differences between the machine language and Assembly language.

| Assembly Language | Machine Language |
| --- | --- |
| Assembly language is an intermediate programming language between a high-level programming language and Machine language | Machine language is a low-level language. |

| | |
|---|---|
| Assembly language is English syntaxes, which is understood by the CPU after converting it to low-level language by interpreter and compilers. | Machine language is in the form of 0's and1's (binary format). One showcases the true/on state while zero depicts the false/off state. |
| Programmers can understand the assembly language, however, CPU cannot. | CPU can directly understand Machine language. No need of compiler or assembler. |
| Assembly language is | Machine code differs |

| | |
|---|---|
| a set of instructions which are the same irrespective of platform. | platform to platform. |
| The codes and instructions of assembly language can be memorized. | Binary codes here can’t be memorized. |
| Modification is not that tough here. | Modification is not possible. It has to be written from scratch for a specific type of CPU. |
| Here applications are device | CDs, DVD’s and BlURAY Discs |

| | |
|---|---|
| drivers, low-level embedded systems, and real-time systems | represent an application of binary for m |