Computer Organization & Assembly Language

(Lecture 01)

Omar Bin Samin

Lecturer

Institute of Management Sciences, Peshawar.

Reference Material

- William Stallings - Computer Organization and Architecture Designing for Performance (8th Edition)

- Kip R. Irvine - Assembly Language for x86 Processors (6th Edition)

- Kip R. Irvine - Assembly Language for Intel based Computers (4th Edition)

Programming Languages

- A programming language is used by a programmer to direct a computer, device or a system to accomplish a specific set of steps which lead to a desired outcome

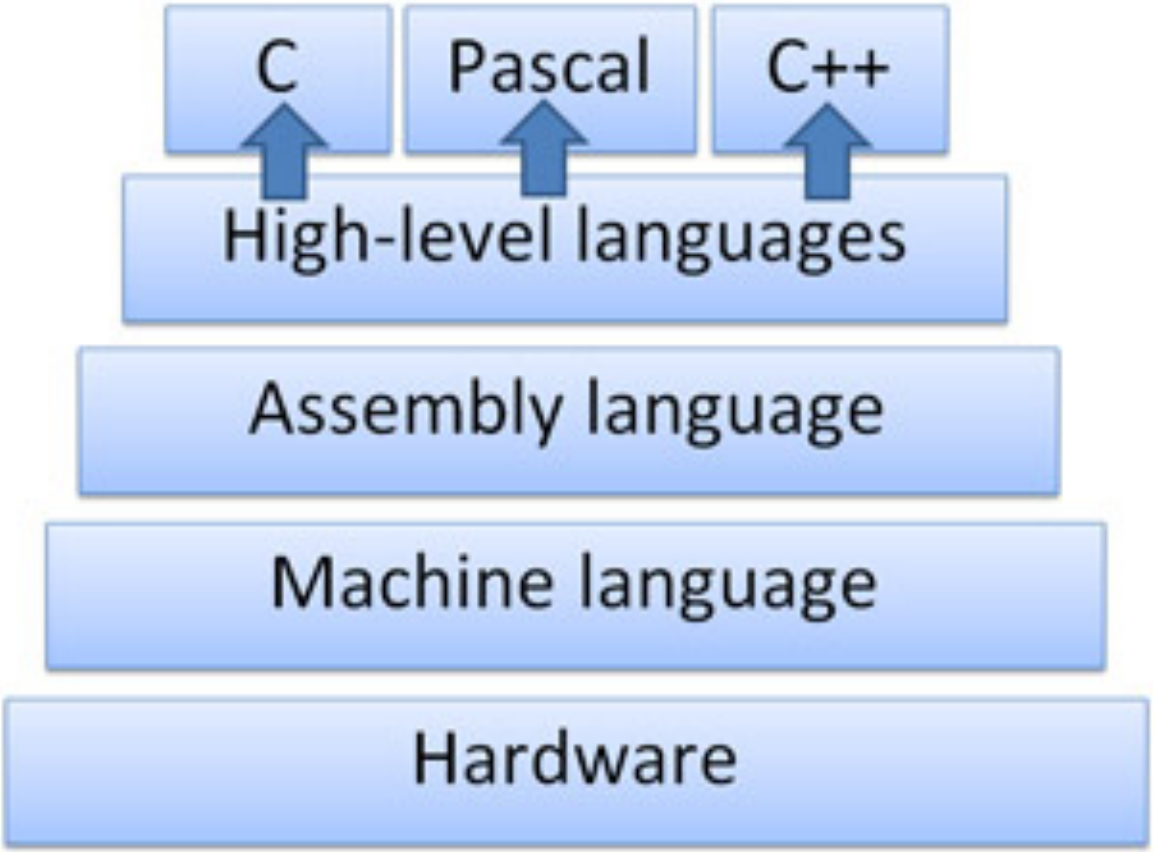
Programming Languages Characterization

- Computer language is a set of instructions

- According to evolutionary scale, generally computer language is divided into the following two main categories:
    - **Low-Level Languages:** A language that corresponds directly to a specific machine

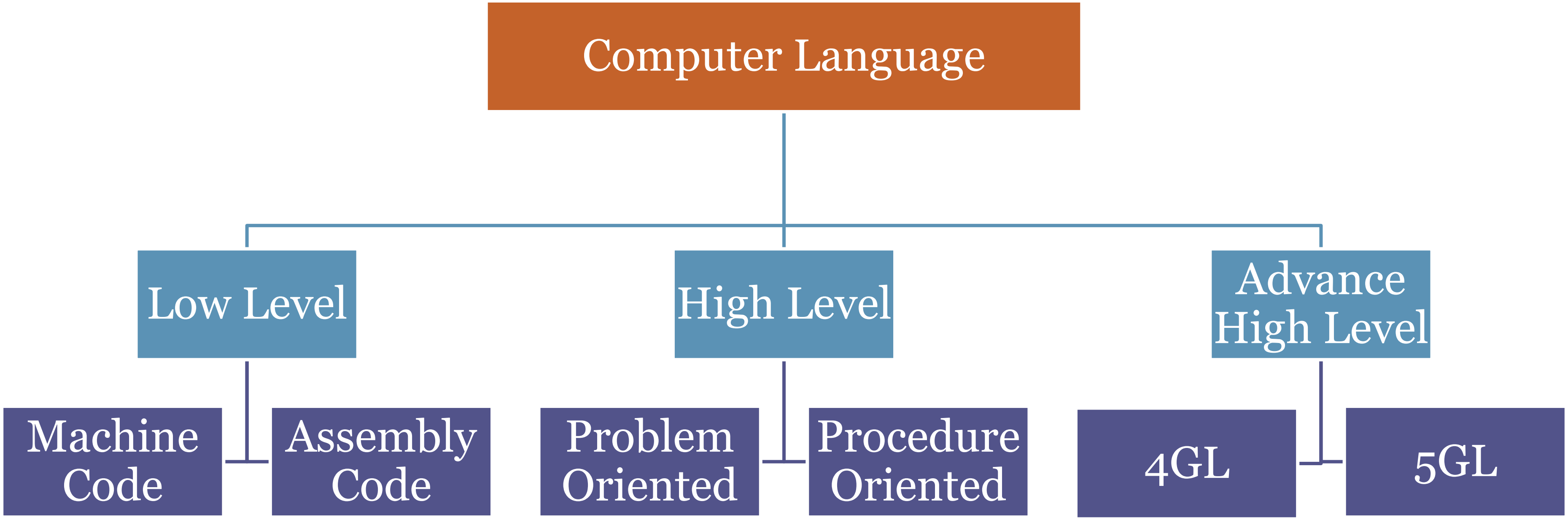
    - **High-Level Languages:** Any language that is independent of the machine

Levels of Computer Programming Languages



- A third type of computer language is also developed, known as **Advance High-Level Language** which contains 4GL and 5GL

- GL stands for Generation Language


Computer Language
Low Level
High Level
Advance High Level
Machine Code
Assembly Code
Problem Oriented
Procedure Oriented
4GL
5GL

Generation Wise Classification

- 1st Generation **(1940-50)**
- 2nd Generation **(1950-1958)**
- 3rd Generation **(1958-85)**
- 4th Generation **(After 1985)**
- 5th Generation **(After 1990)**

1st Generation

- Machine code is the 1st generation language
- It is written in binary (0's and 1's)
- It is a machine oriented and complex language
- It is executed without any language translators
- Machine language is the "native tongue" of the computer, the language closest to the hardware itself

- Each unique computer has a unique machine language

- A machine language program is made up of a series of binary patterns (e.g., 01011100) which represent simple operations that can be accomplished by the computer (e.g., add two operands, move data to a memory location)

- Programming in machine language requires memorization of the binary codes and can be difficult for the human programmer

- **Advantages**
  - Instructions written in binary are immediately executable without the help of any language translators

- **Disadvantages**
  - It is difficult to understand and develop a program using machine language
  - It is a machine-oriented language
  - The knowledge of computer internal architectures is essential for program coding
  - Time consuming coding
  - Debugging is tough and difficult

2nd Generation

- Assembly Language is a low level 2nd generation language
- The machine language instructions are replaced with simple mnemonic abbreviations
- An assembly language program requires translation to machine language. This translation is accomplished by a computer program known as an **Assembler**

- Some common mnemonics are:
  - **ADD** for Addition
  - **SUB** for Subtraction
  - **LDA** for load Accumulator
  - **STA** for store Accumulator

- **Advantages**
  - Coding is faster than machine language because mnemonics are used for program coding
  - Debugging is easy
- **Disadvantages**
  - Machine oriented language
  - The good knowledge of machine architectures is required
  - Translator is used to translate program into machine code
  - Not as fast as machine code language in terms of execution

3^{rd} Generation

- The time and cost of creating machine and assembly languages were quite high and this is a first motivation for development of high-level computer language

- High level language contains a set of instructions written in simple English. It is a set of some symbols, words and rules (syntax) to instruct a machine

Problem Oriented

- A computer language designed to handle a particular class of problem

- Examples
  - **COBOL** (Common Business-Oriented Language) was designed for business, finance and administrative systems

  - **FORTRAN** (Formula Translation) for numeric and scientific computing

Procedure Oriented

- A computer programming language that specifies a series of well-structured steps and procedures within its programming context to compose a program

- It contains a systematic order of statements, functions and commands to complete a computational task or program

- The most common examples of procedural language are C/C++ and Java

- **Advantages**
  - Simple English is used for program coding
  - Machine independent
  - Problem and procedure oriented
  - The knowledge of computer architectures is not necessary
  - It requires less time for program coding
  - Program can be debugged easily and program maintenance is also easy
- **Disadvantage**
  - Translator is used to translate program into machine code
  - Not as fast as machine code language in terms of execution

4th Generation

- It is a high level language in which fewer instruction codes are used to accomplish a particular task
- It is a non-procedural language
- Major 4th generation languages are used to get information from files and databases
- Database oriented programming language
- Handles user queries
- Codes are easy to write, read and understand
- Machine independent language

Non-Procedural Language

- A computer language that does not require writing traditional programming logic

- Also known as a **Declarative Language**

- Users concentrate on defining the inputs and outputs rather than the program steps

- The most common example of non-procedural language is SQL (Structured Query Language). In SQL, only commands are used to implement any task

5th Generation

- Fifth-generation languages are used mainly in artificial intelligence research

- PROLOG (acronym for PROgramming LOGic) is an example of a Logical Programming Language. It uses a form of mathematical logic to solve queries on a programmer-given database of facts and rules