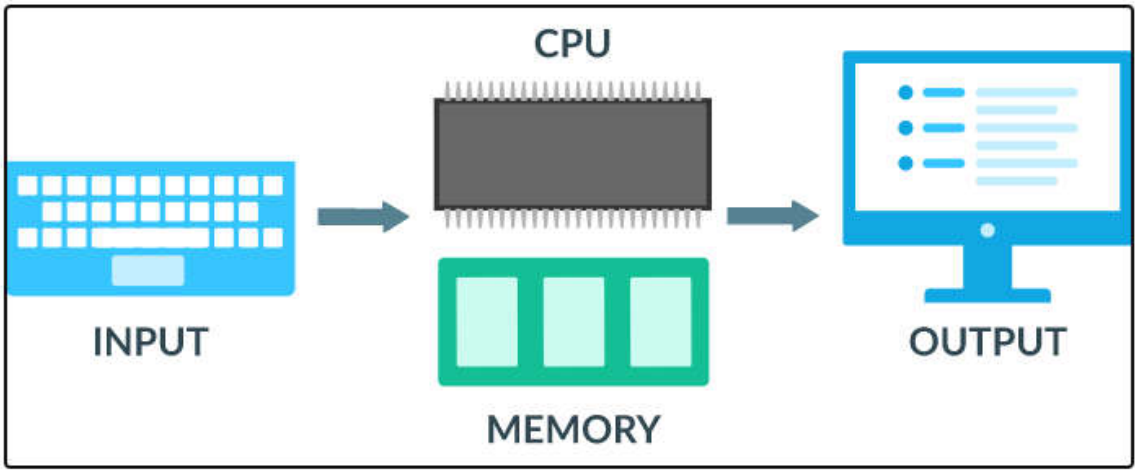
Agenda for Class 2

- **A) Introduction to Programming**
- **B) Using a Flowchart tool**

A) Introduction to Programming

- Before we understand what programming is, we must know what is a computer.
- A computer is a device that can accept human instruction, processes it and responds to it.
- A computer is a computational device which is used to process the data under the control of a computer program.
- Each computer receives **input** from a variety of devices, processes that data with the **CPU** and **memory**, and sends results to some form of **output**.



- The basic components of a computer are:
    - Input Unit
    - CPU – Central Processing Unit
    - GPU – Graphical Processing Unit
    - Storage Unit (Secondary storage)
    - Output Unit
- CPU is called the brain of our computer because
    - it accepts data, (but how??)
    - provides temporary memory space to it until it is stored (saved) on the hard disk,
    - performs logical operations on it and hence processes (here also means converts) data into information.

Activity:

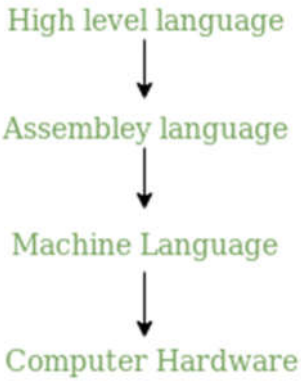
- Examples for input, output devices
- Description about GPUs.

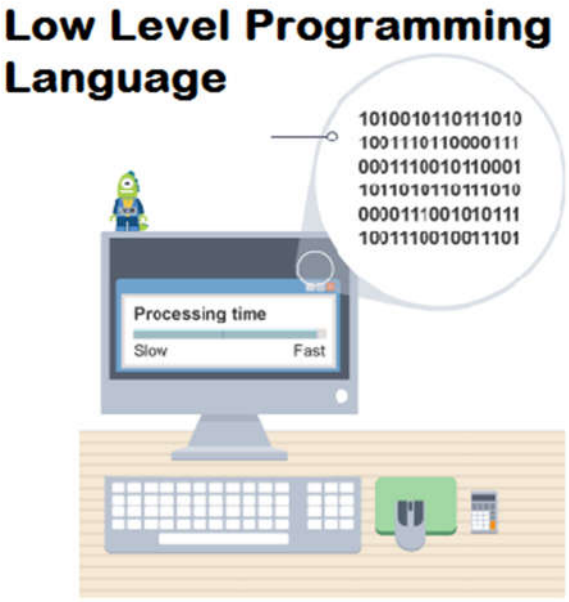
About Programs

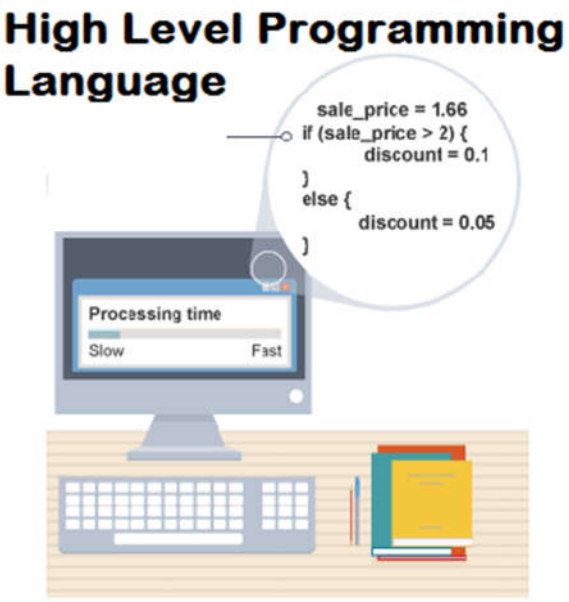
- Program is a **set of instructions** along with data.
- Software is a set of programs that performs multiple tasks together.
    - Eg: An operating system is also a software (system software) that helps humans to interact with the computer system.
    - A program is a set of instructions given to a computer to perform a specific operation.
    - While executing the program, raw data is processed into a desired output format.
    - These **computer programs are written in a programming language** which are high level languages.
        - High level languages are nearly human languages which are understandable by humans (English, Tamil, Hindi, etc.), which are not understandable by computers.
        - Computer understandable language which are called machine language, or low-level language.
        - we have different languages like C, C++, C#, Java, python, etc to communicate with the computers.
        - The computer only understands binary language (the language of 0's and 1's) also called machine-understandable language or low-level language.

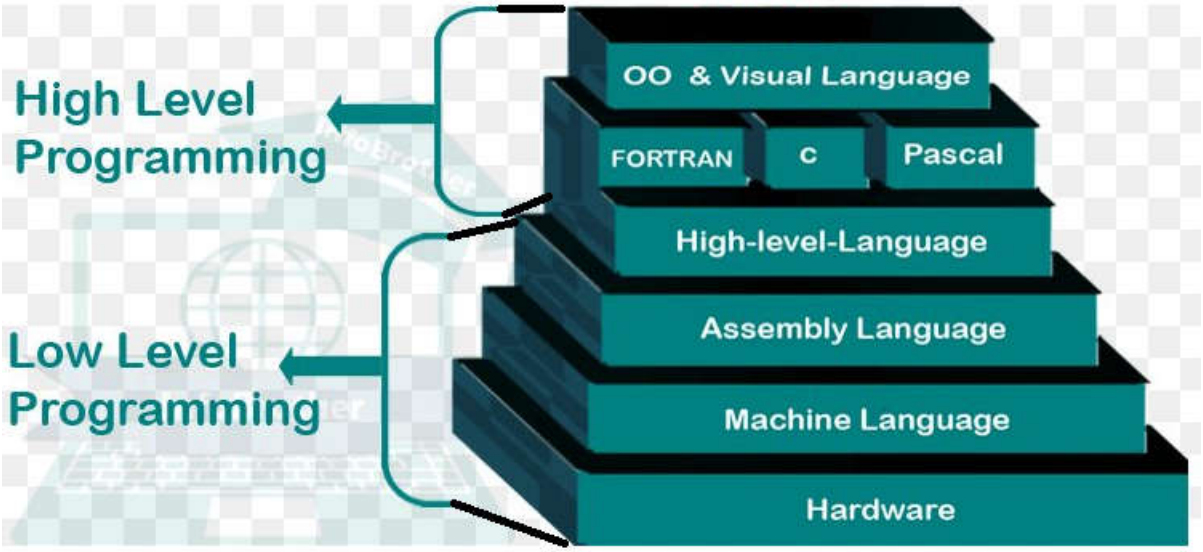
- But the programmers will write programs in high-level languages, which are almost similar to human languages.

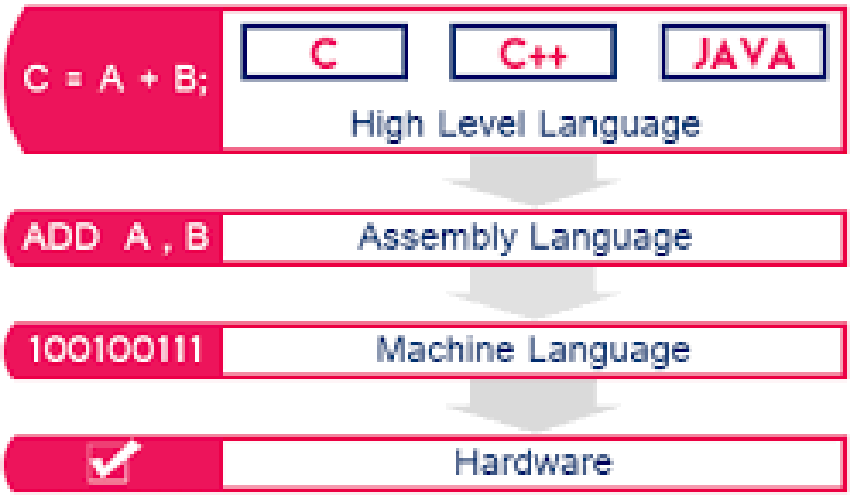
Hierarchy of Computer language

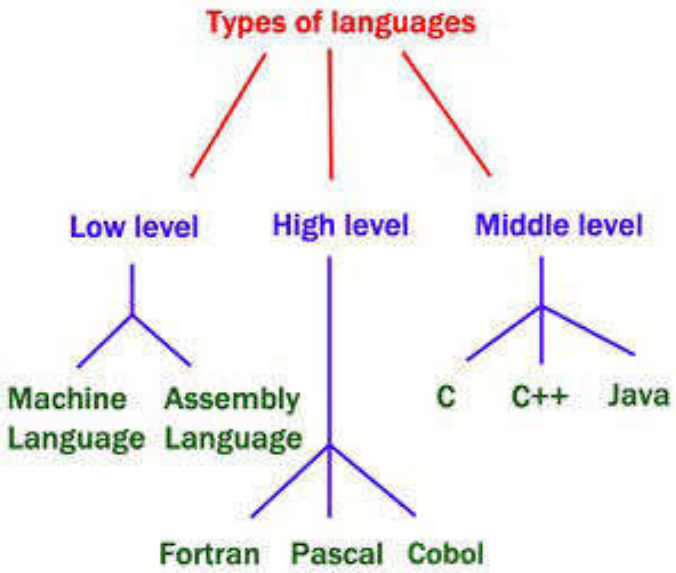


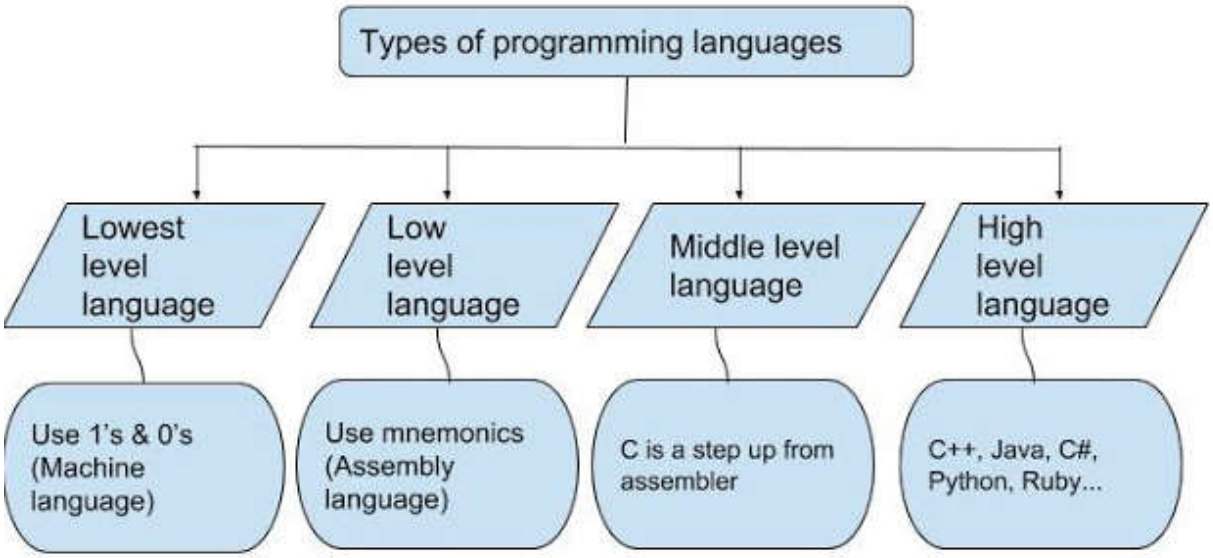


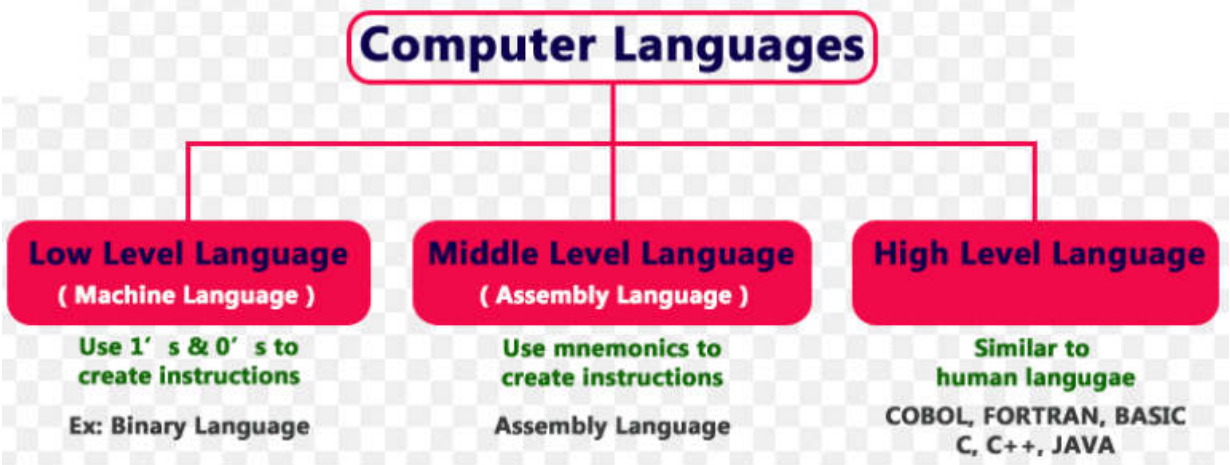


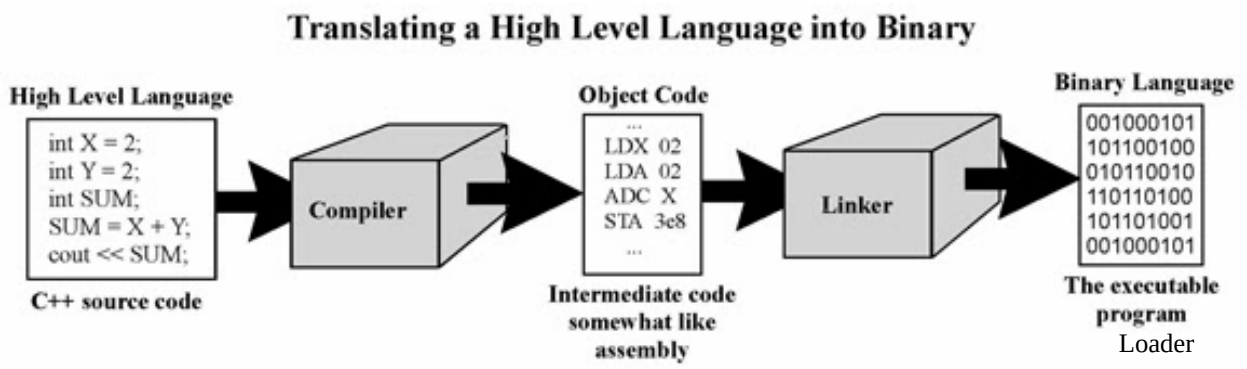

High Level Programming
Low Level Programming
OO & Visual Language
FORTRAN
C
Pascal
High-level-Language
Assembly Language
Machine Language
Hardware


C = A + B;
C
C++
JAVA
High Level Language
ADD A , B
Assembly Language
100100111
Machine Language
Hardware


Types of languages
Low level
High level
Middle level
Machine Language
Assembly Language
C
C++
Java
Fortran
Pascal
Cobol


Types of programming languages
Lowest level language
Low level language
Middle level language
High level language
Use 1's & 0's (Machine language)
Use mnemonics (Assembly language)
C is a step up from assembler
C++, Java, C#, Python, Ruby...


Computer Languages
Low Level Language
(Machine Language)
Use 1' s & 0' s to create instructions
Ex: Binary Language
Middle Level Language
(Assembly Language)
Use mnemonics to create instructions
Assembly Language
High Level Language
Similar to human langugae
COBOL, FORTRAN, BASIC
C, C++, JAVA



S.NO	HIGH LEVEL LANGUAGE	LOW LEVEL LANGUAGE
1.	It is programmer friendly language.	It is a machine friendly language.
2.	High level language is less memory efficient.	Low level language is high memory efficient.
3.	It is easy to understand.	It is tough to understand.
4.	It is simple to debug.	It is complex to debug comparatively.
5.	It is simple to maintain.	It is complex to maintain comparatively.
6.	It is portable.	It is non-portable.
7.	It can run on any platform.	It is machine-dependent.
8.	It needs compiler or interpreter for translation.	It needs assembler for translation.
9.	It is used widely for programming.	It is not commonly used now-a-days in programming.

There have been many programming languages some of them are listed below:

C	Python	C++
C#	R	Ruby
COBOL	ADA	Java
Fortran	BASIC	Altair BASIC
True BASIC	Visual BASIC	GW BASIC
QBASIC	PureBASIC	PASCAL
Turbo Pascal	GO	ALGOL
LISP	SCALA	Swift
Rust	Prolog	Reia
Racket	Scheme	Shimula
Perl	PHP	Java Script
CoffeeScript	VisualFoxPro	Babel
Logo	Lua	Smalltalk
Matlab	F	F#
Dart	Datalog	dbase
Haskell	dylan	Julia
ksh	metro	Mumps
Nim	OCaml	pick
TCL	D	CPL
Curry	ActionScript	Erlang
Clojure	DarkBASCIC	Assembly

Characteristics of a programming Language

- A programming language must be
    - simple,
    - easy to learn and use,
    - with good readability and
    - human recognizable.
- Programming language's **efficiency** must be high so that it can be easily converted into a machine code and executed consumes little space in memory. (ATM machine delay irritates customers)
- A programming language should be
    - well-structured and
    - documented
    - so that it is suitable for application development.
- Necessary tools for
    - development,
    - debugging,
    - testing,
    - maintenance of a program must be provided by a programming language.
- A programming language should provide single environment known as Integrated Development Environment (IDE).
- A programming language must be consistent in terms of syntax and semantics.

Example: A simple c program

- The piece of code given below performs a basic task of printing “hello world! I am learning programming” on the console screen.
- We must know that keyboard, scanner, mouse, microphone, etc are various examples of input devices
- and monitor (console screen), printer, speaker, etc are the examples of output devices.

```
main()
{
clrscr();
printf(“hello world! I am learning to program”);
getch();
}
```

At this stage, you might not be able to understand in-depth how this code prints something on the screen.

- The main() is a standard function that you will always include in any program.
- The execution of any program starts from the main() function.
- The clrscr() function is used to clear the screen.
- The printf () function helps us to print the desired output on the screen.
- Also, getch() is a function that accepts any character input from the keyboard.

- In simple words, we need to press any key to continue (some people may say that getch() helps in holding the screen to see the output).

Sample 'C' Programs:

C Version

Printing a statement
#include <stdio.h>
int main()
{
printf ("Welcome to our VIT....");
return 0;
}

Adding Two numbers and print the sum
 	
`#include <stdio.h>`
`int main()`
`{`
`  int x, y, z;`
 	
`  printf("Enter two numbers to add\n");`
`  scanf("%d %d", &x, &y);`
 	
`  z = x + y;`
 	
`  printf("Sum of the numbers = %d\n", z);`
 	
`  return 0;`
`}`

Activity:

- To use online C compiler to practice the following C programs
  - *Adding 3 numbers*
  - *Calculating the total and average marks of a student with six subjects.*
  - *To find the largest among three given numbers, which are different from each other.*
- To use online tool (www.draw.io) to draw the flow chart for the above C programs