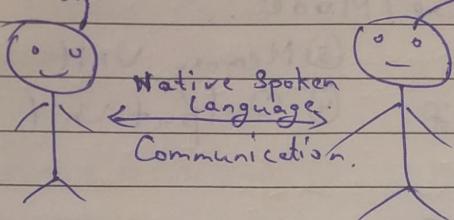
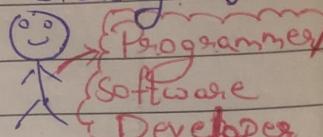


What is the result
 of adding 2 & 3?

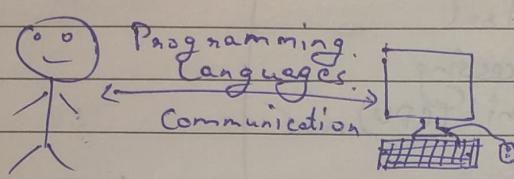


The result is
 5.

Information Technology [IT]
 Industry



Programming Lang
 C, C++, Java, Python



PROGRAM

Software
 Application
 APP

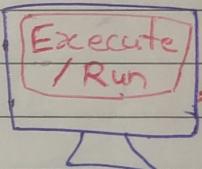
Instruction 1
 Instruction 2
 Instruction 3

PL
 Grammar
 Syntax

English
 Alphabets
 Charset

A Computer is an electronically
 programmable device.

A program [Computer] is set of
 instruction that is given as
 input to the Computer to
 perform a certain task.



DUMB

Computer.
 Output.

A Software or an application is a collection
 of many programmes

Computers

Fixed - Program
 Computers.

Stored - Program
 Computers.

Eg. Calculators,
 X-ray Machine,
 Washing Machine, etc

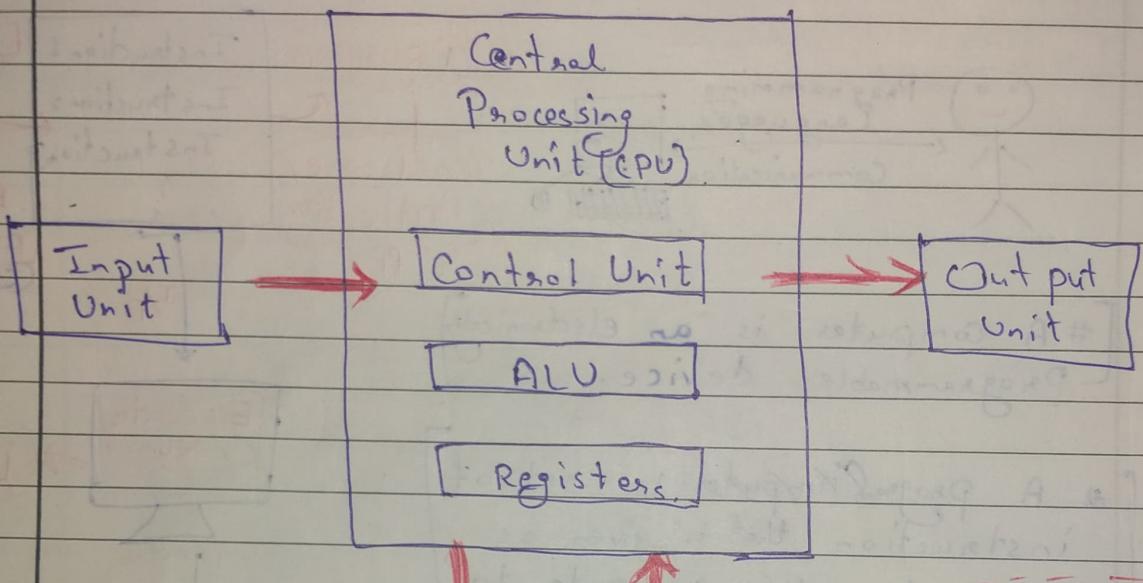
Eg. Modern Day Computers
 Desktops, PCs, Laptops,
 Mobile, etc

John Von Neumann

Architecture / Model.

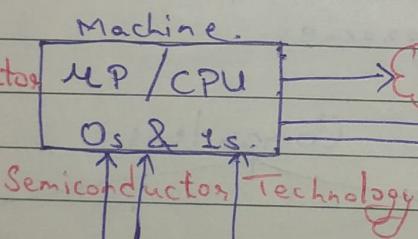
- ① Input Unit
- ③ Memory Unit
- ② Processing Unit
- ④ Output Unit

Computer Organization.



1940s.

Conductors & Insulators
Semiconductors



Transistors.

PNP

NPN.

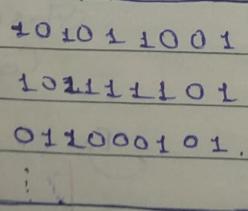
Voltages.

0v
low
0

+5v
high.
1.

1940s.

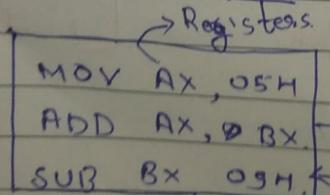
Machine level - language [ML]
Low-level language [LL]



ASSEMBLER

1950s.

Assembly LL [ALL]



COMPILER /
INTERPRETER

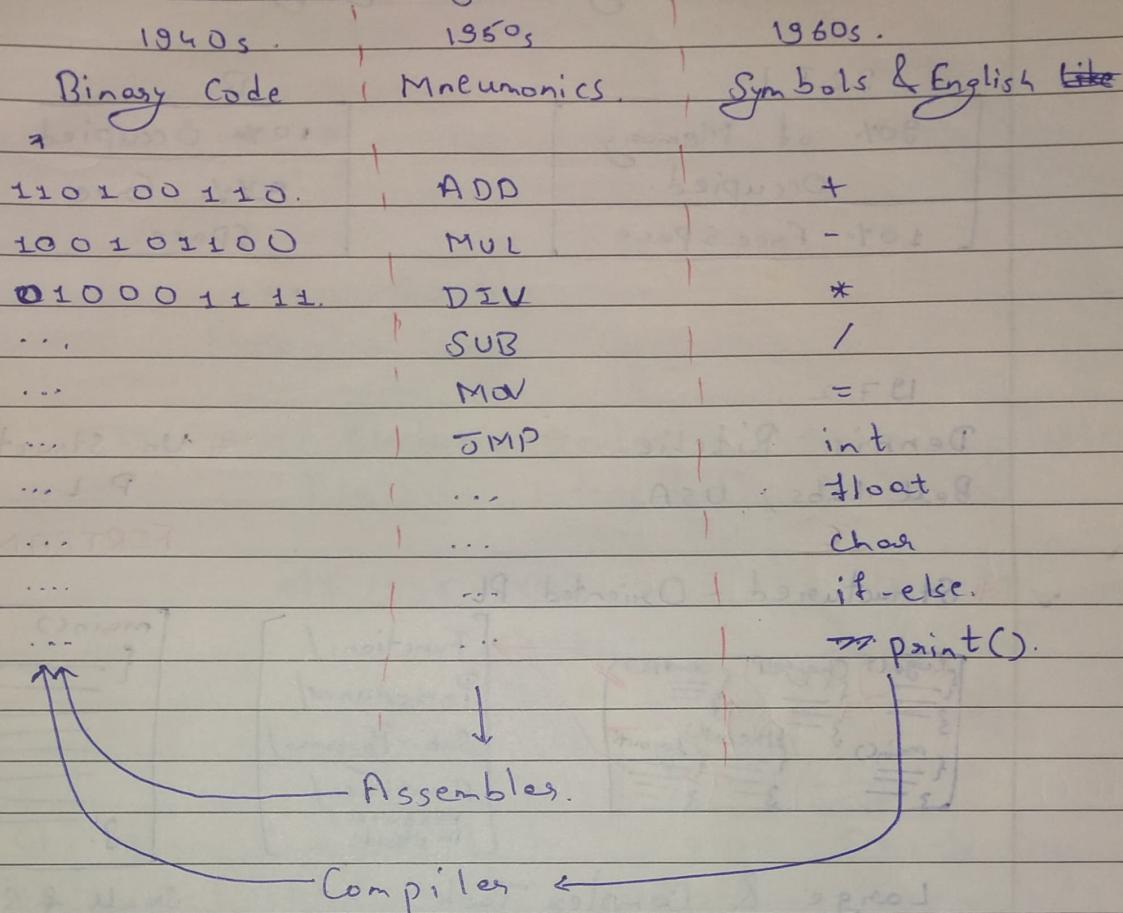
1960s

High-Level Language
C, C++, JS, [HLL], Java, etc.

```

int a = 5;
int b = 9;
int c = a + b;
print(c);
  
```

Mnemonics are the names associated with the Binary instruction.



An assembler is system software which takes assembly level language as input & converts it into machine level language.

A compiler is a system software which takes high level language code as input & converts it into MLL.

Single Block (function).

Programming Paradigms:-

- ① Unstructured / Non- Structured style.
- ② Structured / Procedure style.
- ③ By object - Oriented Programming style
↳ Classes & Objects

Whenever the technologies of two devices with a computer are going to mismatch. In that case the overall speed & efficiency of your entire computer system is going to be as fast as the slowest executing device.

Popular High-Level Languages.

B C P L

90% of Memory Occupied.
10% Free Space.

B.

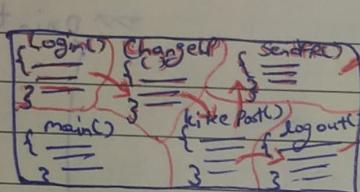
10% Occupied
90% Free Space.

1972.

Dennis Ritchie.
Bell Labs, USA.

"C"

Structured - Oriented PL.



Functions /
Procedures /
Sub-Programs /
Routines /
Modules /
Methods...

main()

10 K
lines of
code

Un-structured PL.

FORTRAN

Large & Complex Task.
Functions ↑
Data. ↓

Small & Simple task.

1979.

Bjarne Stroustrup
Bell Labs, USA

C [POP]

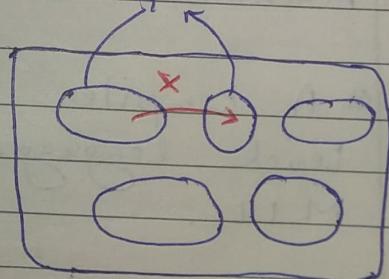
Simula [OOP]

"C with Classes" *

1985 - C++

Hybrid.

Objects.



Data ↑
Functions. ↓

Components of any
Program!
① Data
② Functions.

OOP

int a = 5;

① Classes

++ a;

② Objects.

paint(a); // 6.

++ : Incrementation

1990s

"Internet" - WWW

Sun Microsystem, Inc.
California, USA.

James Gosling,

11 members Team :- Green Team.

TV sets / Electronic Devices - Internet compatible
"Platform independent" ~~not~~

1995 - Oak.

C++ --

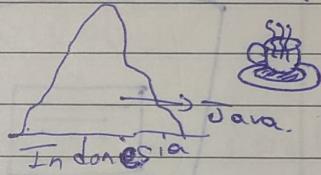
Green

Silk.

JAVA

Netscape Browser

"Java".



- ① Platform Independent.
- ② Object-Oriented.
- ③ Freely available.
- ④ Open source.

2010 - Oracle Corp.

- ① Platform Independent
- ② Object-Oriented
- ③ Freely available
- ④ Closed sources.

Software

System Software. [Hardware]

Speed of Execution ↑

Platform independent ↓

Compiler,

Assembler,

Interpreter,

Operating System, etc.

c/c++

Application Software [User-task]

Platform independent ↑

Speed of execution ↓

LinkedIn,

WhatsApp,

Instagram,

Facebook, etc.

Java/ Python, etc.

1950s.

(HDD) Hard Disk Drive

1956

IBM.

Adv.

- ① Non-Volatile
 - ② Inexpensive
 - ③ More Storage Capacity
- Electro - Mechanical Technology Devices

(Collection of wires).

→ → → Bus → →

Microprocessor / CPU. OS & LS.

Disadv.

- ① Slow.
- ② Bulky.
- ③ Noisy

Semi-Conductor Technology Device.

[Transistors],

1960s

RAM.

1966.

IBM

Adv.

- ① Fast
 - ② Compact
 - ③ No noise.
- Semi-Conductor Technology Device.
[Transistors & Capacitors]

Bus.

Semi-Conductor Technology Device
[Transistors]

- Disadv.
- ① Volatile
 - ② Expensive.

(3) Less Storage Capacity

Semiconductor Technology.

Cache.
8 MB
32 MB
Frequently Requested Data

{ Speed/Cost }
CPU > RAM > HDD.
Storage
Secondary Memory.
Hard Disk Drive

Main memory / Primary memory
RAM

Loading.

saving.

TB

Byte.

GB

+5v

Microprocessor / CPU.
Data.
OS & LS.
Registers.

Semi-Conductor

Technology Device
[Transistors]

Output Unit

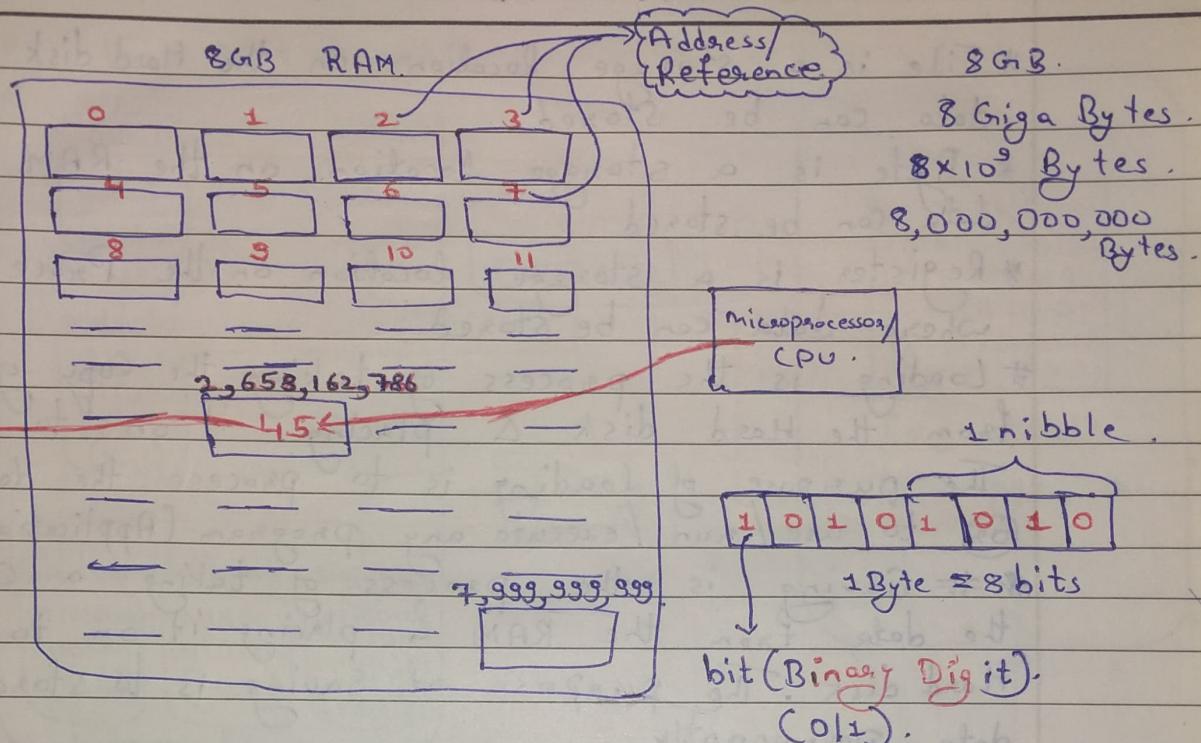
SSD.

Semi-Conductor
Flash Drives

Electro - Mechanical
Technology Device.
(Magnetic Storage)

Semiconductor
Technology Device.
[Transistors & Capacitors]

Operating System [OS]



Memory Measurement Units:-

$$1 \text{ nibble} = 4 \text{ bits}$$

$$1024 \text{ TB} = 1 \text{ PB} \text{ (Petabytes)}$$

$$1 \text{ Byte} = 8 \text{ bits}$$

$$1024 \text{ PB} = 1 \text{ EB} \text{ (Exabytes)}$$

$$1024 \text{ Bytes} = 1 \text{ KB}$$

$$1024 \text{ EB} = 1 \text{ ZB} \text{ (Zettabytes)}$$

$$1024 \text{ KB} = 1 \text{ MB}$$

$$1024 \text{ ZB} = 1 \text{ YB} \text{ (Yottabytes)}$$

$$1024 \text{ MB} = 1 \text{ GB}$$

$$1024 \text{ YB} = 1 \text{ Bronto Byte}$$

$$1024 \text{ GB} = 1 \text{ TB}$$

$$1024 \text{ Bronto Bytes} = 1 \text{ Geop Byte}$$

[1 Geop Byte is Highest memory].

We have two types of memory devices within a computer, because we have expectations from the memory units. They are:-

- ① Extremely fast in Execution.
- ② Non-Volatile.
- ③ In-Expensive
- ④ Compact in size.
- ⑤ Less Noisy.
- ⑥ More Storage Capacity.

There is not a single memory device that can satisfy all the six expectations of a computer, memory. Hence, we have two memory devices in a Computer, namely.

- ① Primary Memory also known as RAM / Main memory.
- ② Secondary memory also known as Hard disk [HDD].

* File is a storage location on the Hard disk where data can be stored.

* Byte is a storage location on the RAM where data can be stored.

* Register is a storage location on the Processor where data can be stored.

Loading is the process of taking the Copy of the data from the Hard disk & placing it on to the RAM.

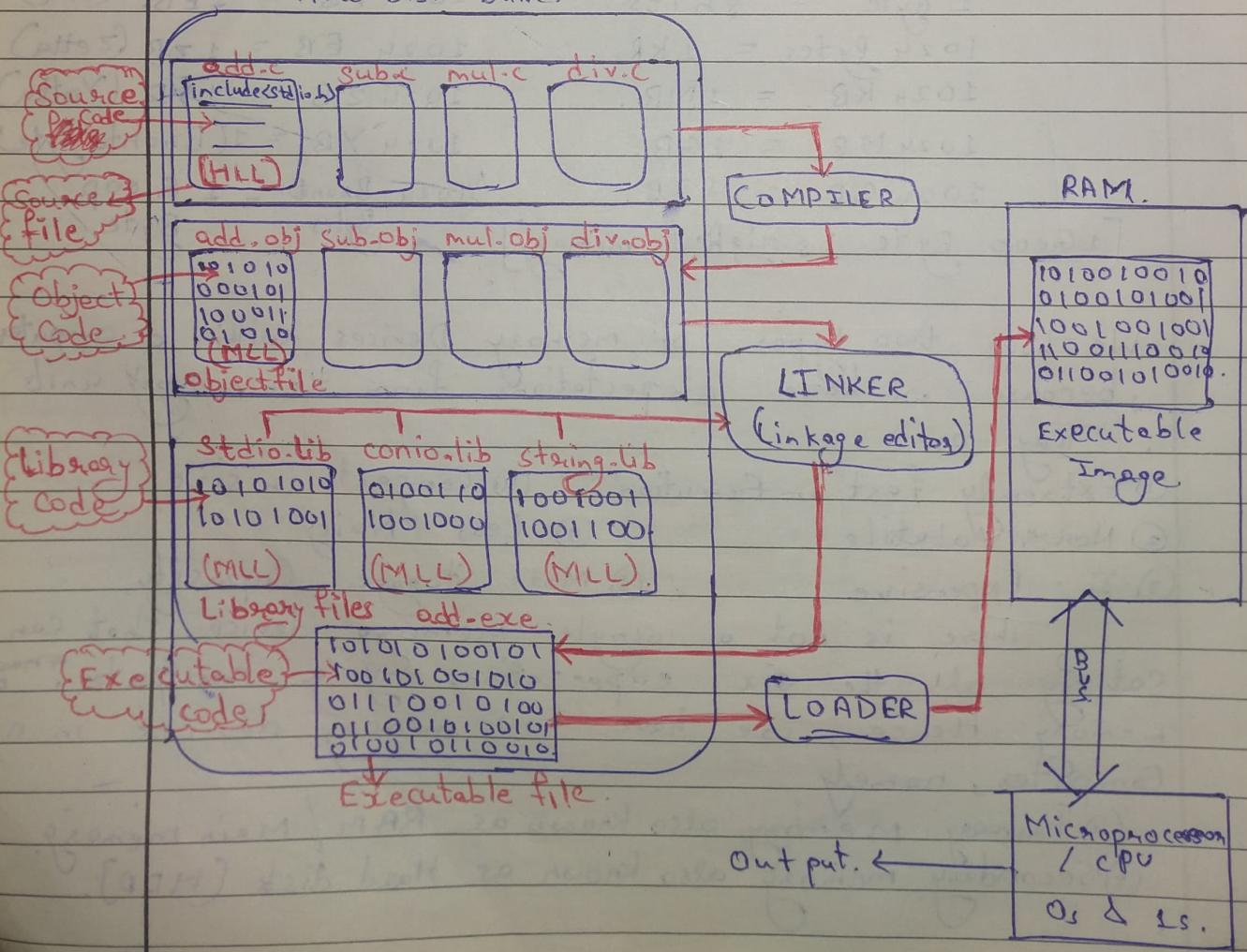
The purpose of loading is to process the data.

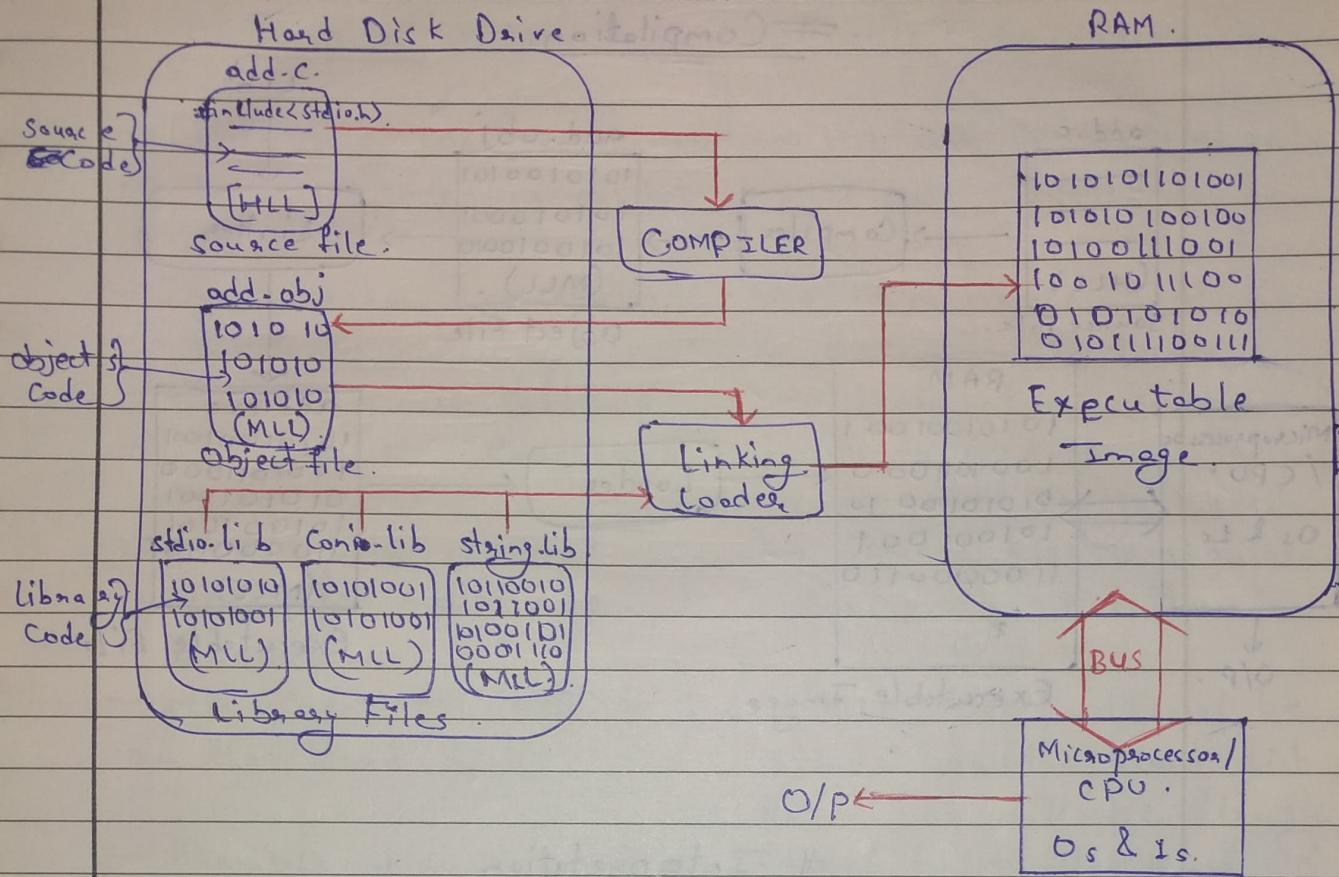
(Qn) to use/run/execute any program [Application]

Saving is the process of taking a copy of the data from the RAM & placing it on to the Hard disk. The purpose of Saving is to store the data permanently.

* A RAM is a collection of bytes.

Hard Disk Drive.





Header File (.h) Vs Library Files (.lib)

stdio.lib

```
int add(int a, int b);
{
    return a+b;
}
```

Function Definition.
[MLL]

stdio.h

```
int add(int, int);
```

Function Prototypes.
[HLL].

`#include <stdio.h>`

`void main()`

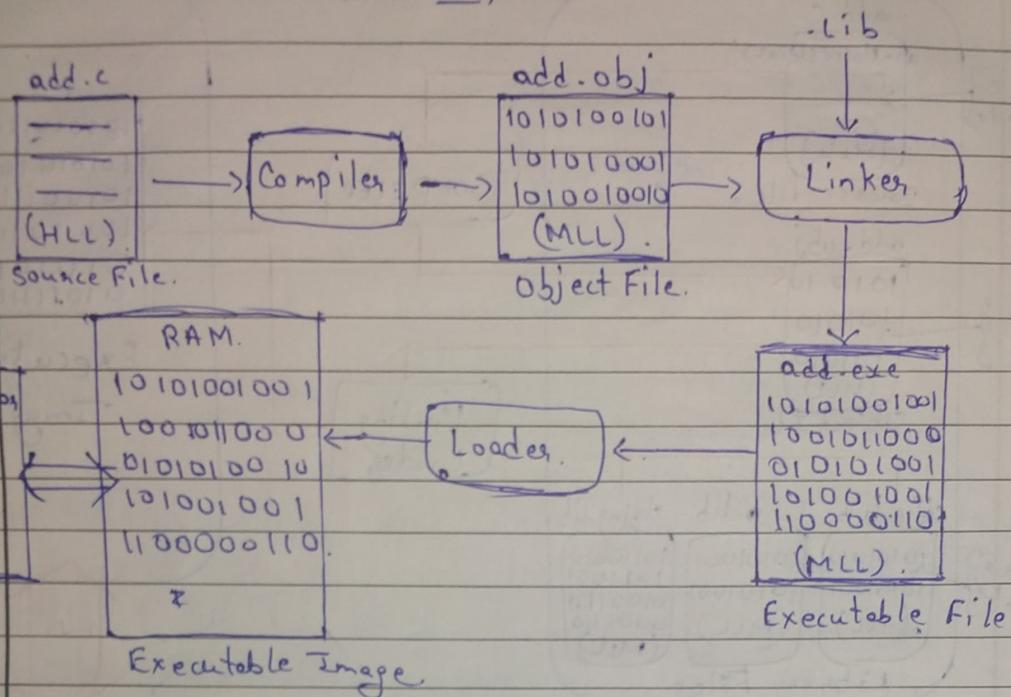
{

`int sum = add(10, 20);`

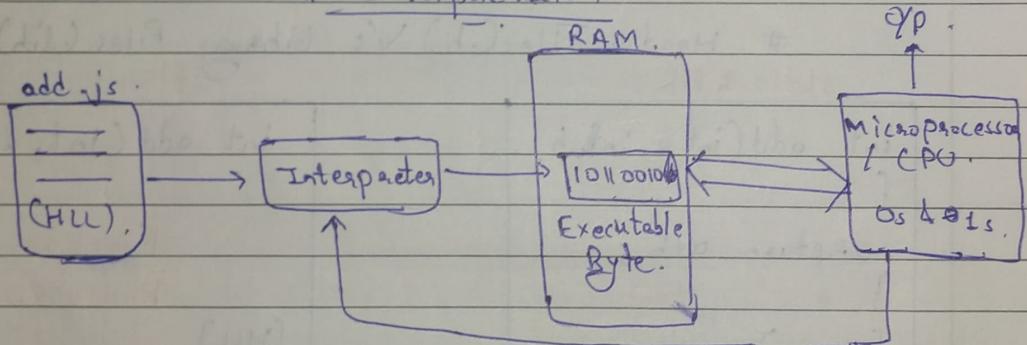
}

Function Call.

Compilation.



Interpretation



Compilation is the process of converting HLL code into MLL code using Compiler as the software.

During compilation process, the entire HLL code is converted into MLL code in one go.

Interpretation is the process of Converting HLL code into MLL code using Interpreter as a software. During Interpretation process, each HLL instruction is converted into MLL code instruction by instruction (one by one)

Compiler

- Entire HLL code is converted → HLL code is converted into MLL code in one shot
- The process of execution is faster.
- Debugging is difficult
- Intermediate Code will be generated.
- Source file is not needed for multiple Execution
- MLL code will be stored on the Hard disk.
- Example of pure compiled language:- C, C++, C#, etc.

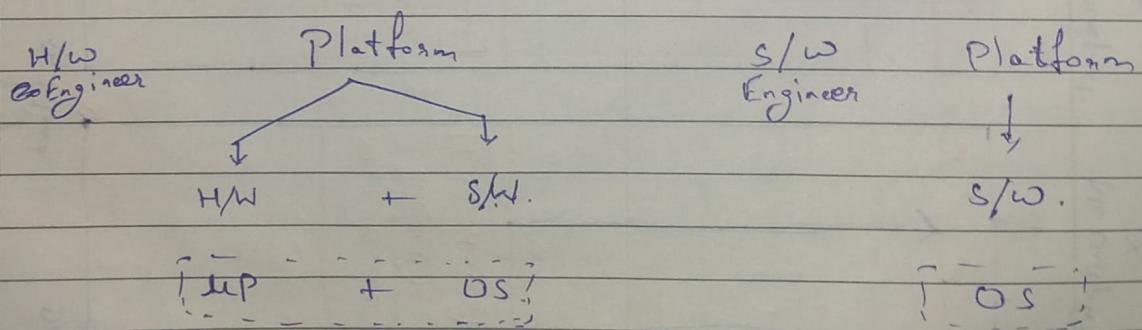
Interpreter

- MLL code instruction by instruction, one at a time.
- The process of execution is slower.
- Debugging is easy
- Intermediate Code will not be generated
- Source file is required for every execution.

- MLL code will not be stored on the Hard disk
Rather, it will be directly loaded on to the RAM.
- Example:- JS, Ruby, PHP, Perl, etc.

Note :- Java is a hybrid programming language as it is both compiled & Interpreted.

Platform Independent



Eg:-

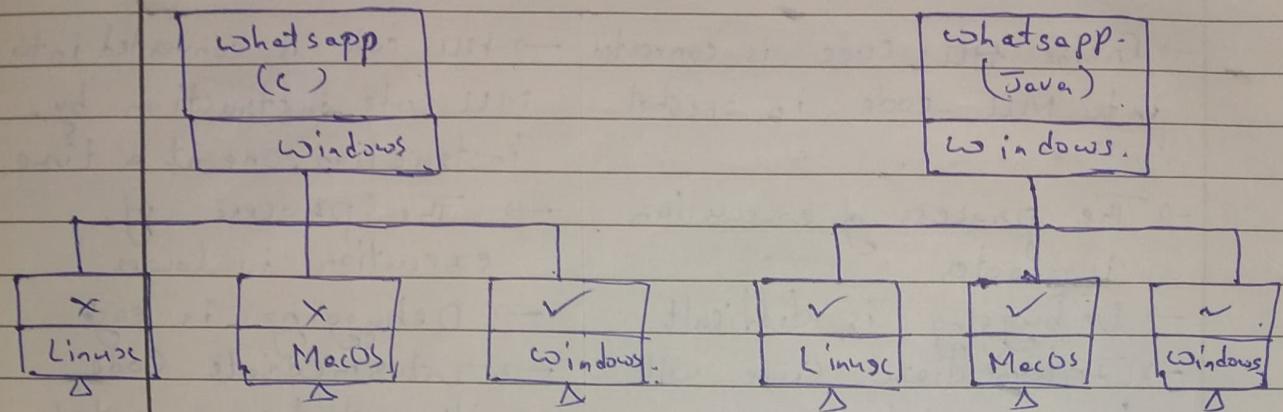
intel i9 + Windows 10.
AMD Ryzen + Linux.
IBM Power10 + Macintosh OS.

Eg:-

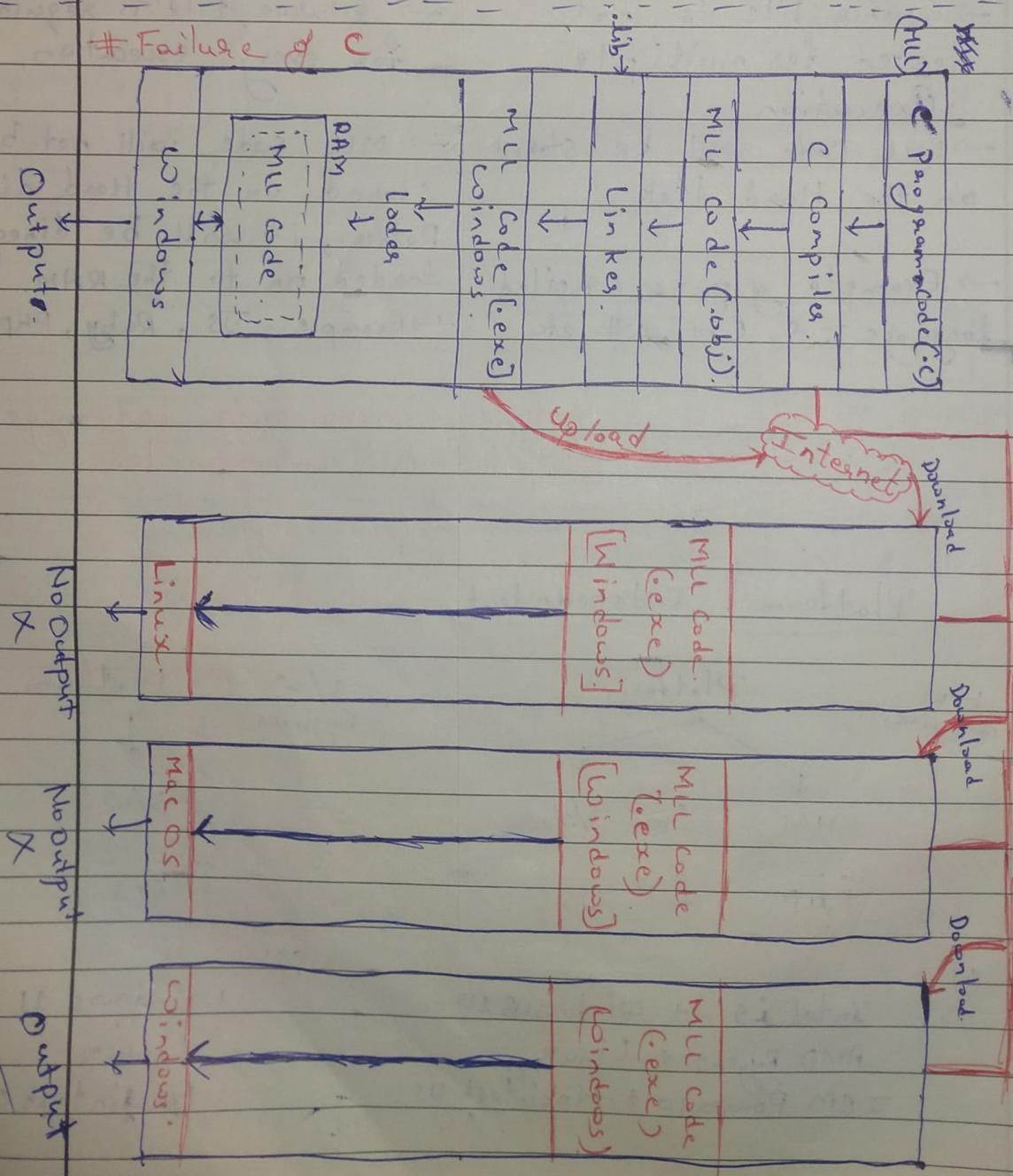
Windows 11,
Linux
Macintosh OS,

Platform Dependency.

Platform Independence.



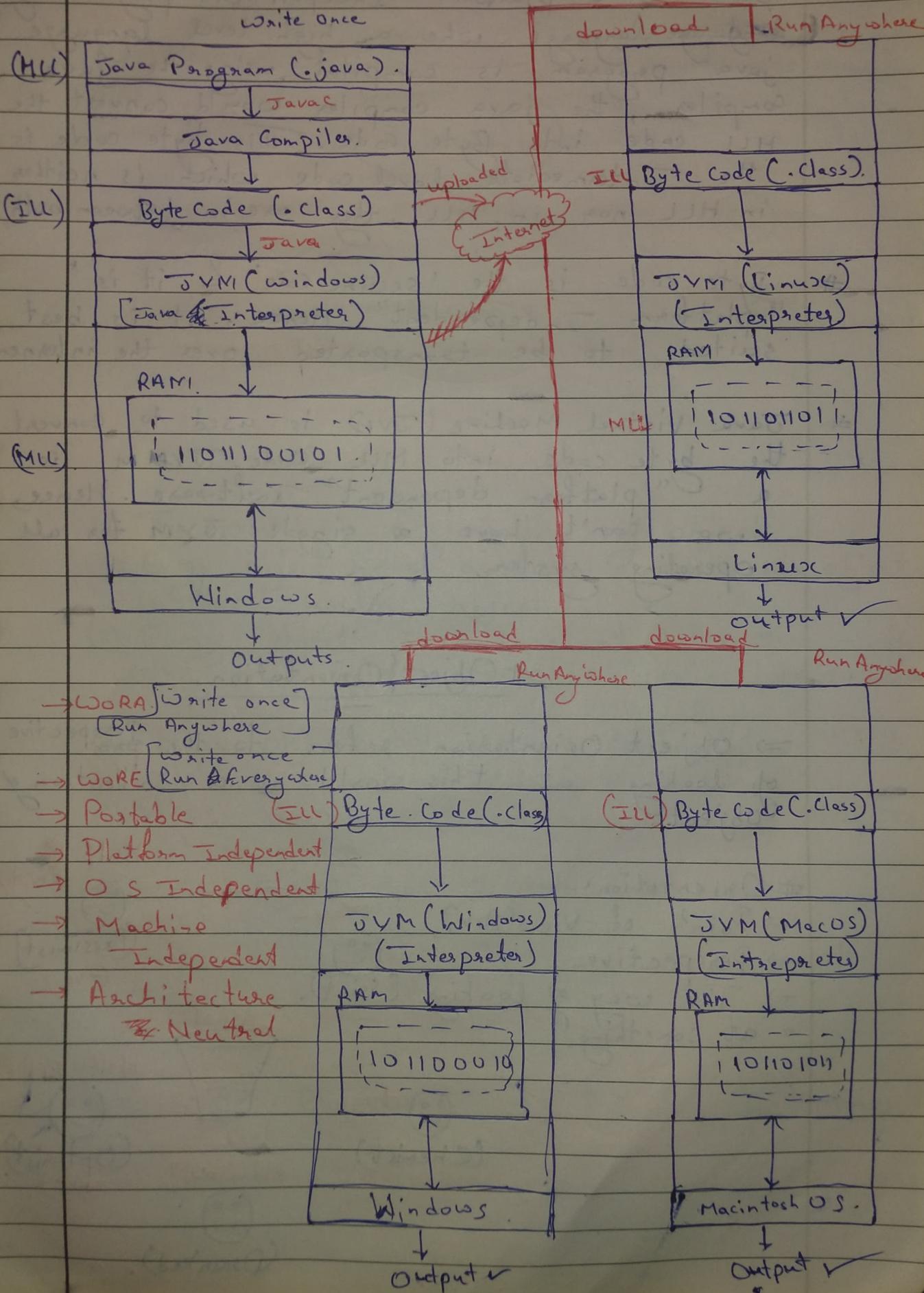
Failure of C.



(III)
 ① Secure Code
 ② Platform Independent
 # success of Java

Page No: _____
 Date: _____

P
W



Java is a platform independent programming language language. When a high level language java program is compiled using the java compiler, the java compiler would convert the HLL code into Byte Code. This byte code is the Intermediate level code which is neither in HLL nor in MLL, they are in between both.

Byte code is the 'secure code' & it is "platform independent". Hence, it is best suited to be transported over the internet.

Java Virtual Machine (JVM) is used to convert the byte code into MLL Code. JVM is a "platform dependent" software. Hence, we can't have a single JVM for all operating systems.