Java Programming

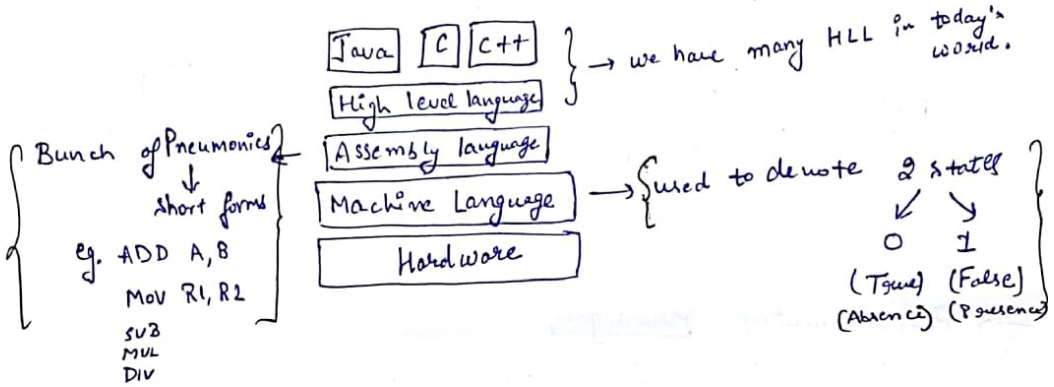
(Lecture - 1)

Programming language : Set of instructions for a computer.

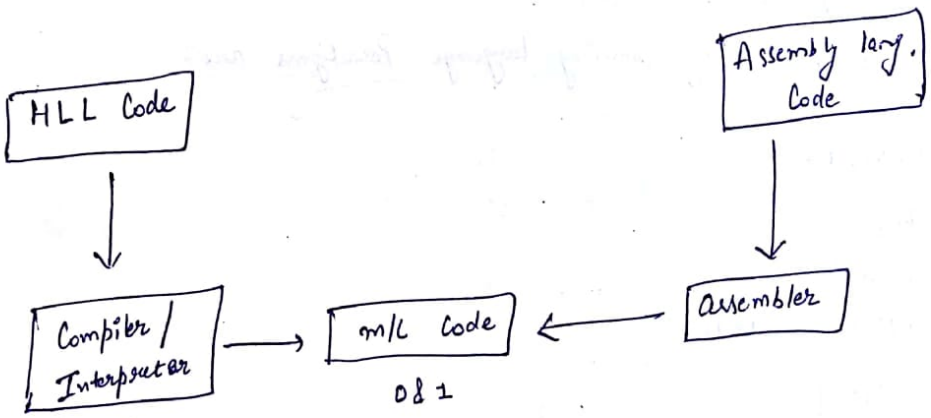
A formal computer language designed to communicate and instruct a m/c particularly a computer.

It controls the behavior of a computer device.

⇒ Levels of PL :



1. Machine level (m/c lang.)
2. Low level (assembly lang)
3. High level (C, C++, Java, etc.)



Compiler, Interpreter & Assembler

↓↓

Language Processors

Interpreters: the lang. processor that translates/convert each statement of source program into machine code and executes it immediately before translating the next statement.

Compilers: The language processor that translates the complete source program as a whole in machine code before execution.

Assemblers: It is used to translate the program written in Assembly language into m/c code.

⇒ Programming Paradigms

↓

means A world view

→ style/way of programming

Paradigm – A pattern/model in which we try to accomplish a task.

Major Programming language Paradigms are:

1. Imperative
2. Logical
3. functional
4. Object Oriented

Imperative

The imperative programming paradigm assumes that the computer can maintain through environments of variables any changes in a computation process.

computations are performed through a guided sequence of steps, in which these variables are referred to or changed. The order of the steps is crucial.

Imperative languages :

Advantages : – efficient
– close to the machine
– popular
– familier

Disadvantages :

– Semantics of a program can be complex to understand or prove because referential transparency does not hold.
– Side effects also make debugging harder.
– Abstraction is more limited.
– Order is crucial

(2) Logical

The Logical Paradigm takes a declarative approach to problem Solving. Various logical assertions about a situation are made, establishing all known facts. Then queries are made. The role of computer becomes maintaining data and logical deduction.

A logical program is divided into three sections:
1. A series of definitions / declarations that define the problem domain.
2. Statements of relevant facts.
3. statement of goals in the form of a query.

Any deductible solution to a query is returned.

Advantages

1. The system solves the problem, so the programming steps themselves are kept to a minimum.
2. Proving the validity of a given program is simple.

③ Functional

↳ It views all subprograms as functions in the mathematical sense - informally, they take in arguments and return a single solution.

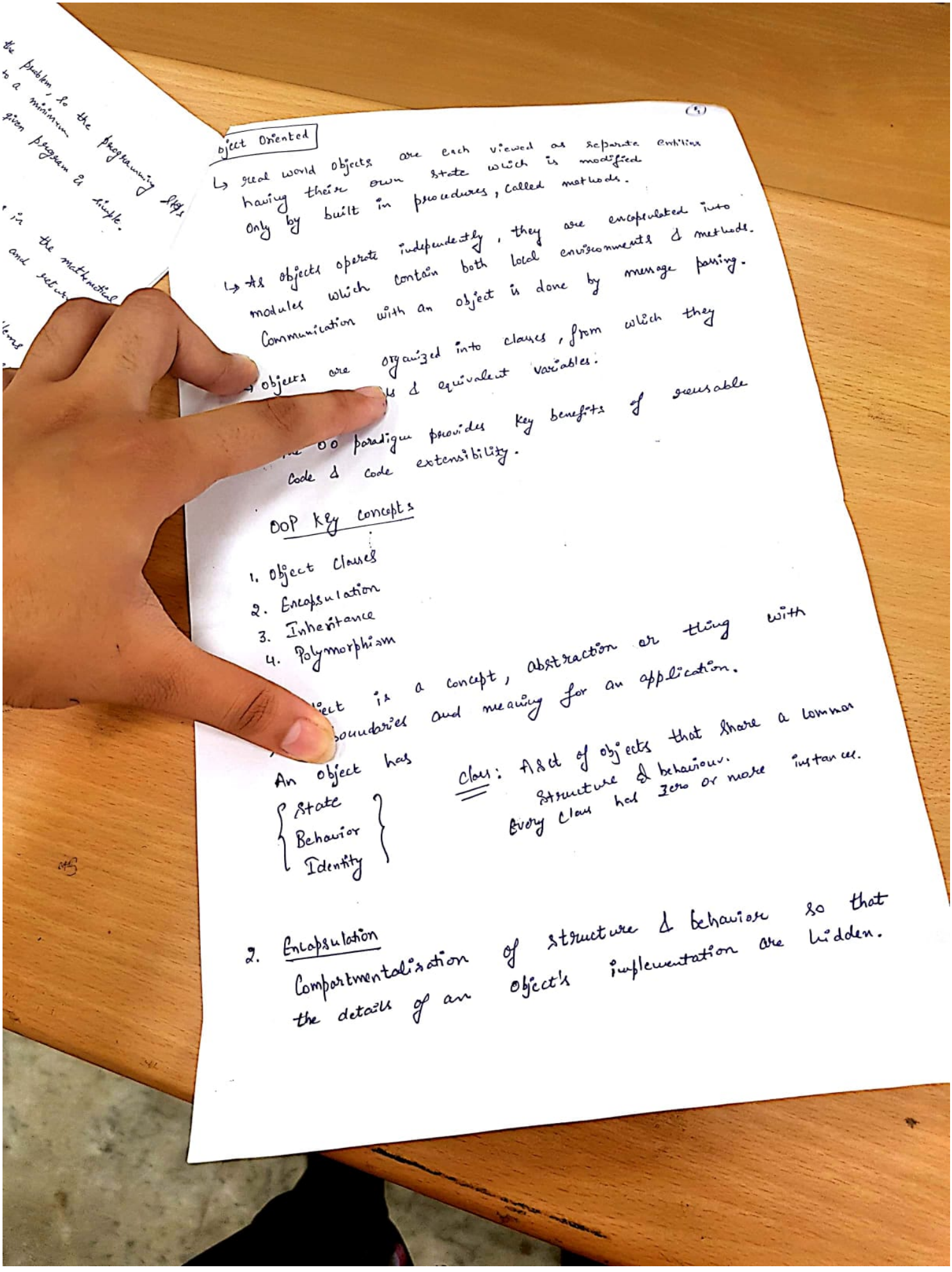
Functional languages permit functional solutions to problems by permitting a programmer to treat functions as first class objects.

Advantages

1. The high level of abstraction, especially when functions are used, supresses many of the details of programming & thus removes the possibility of committing many classes of error.
2. Evaluation order independence (lack of dependence on assignment opr.) It makes functional oriented languages good candidates for programming massively parallel computers.

Disadvantages

1. less efficiency
2. Problems involving many variables or a lot of sequential activity are sometimes easier to handle imperatively or with object-oriented programming.



(1)

Object Oriented

↳ Real world objects are each viewed as separate entities having their own state which is modified only by built in procedures, called methods.

↳ As objects operate independently, they are encapsulated into modules which contain both local environments & methods. Communication with an object is done by message passing.

→ objects are organized into classes, from which they ... & equivalent variables.

... OO paradigm provides key benefits of reusable code & code extensibility.

OOP key concepts

1. Object Classes
2. Encapsulation
3. Inheritance
4. Polymorphism

...ject is a concept, abstraction or thing with ...boundaries and meaning for an application.

An object has
- State
- Behavior
- Identity

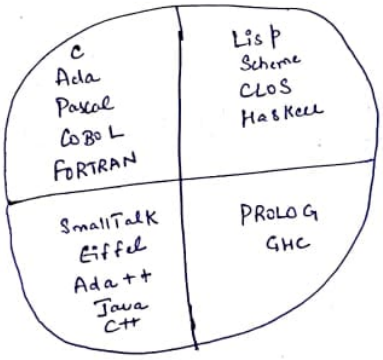
Class: A set of objects that share a common structure & behaviour. Every class has zero or more instances.

2. Encapsulation

Compartmentalisation of structure & behaviour so that the details of an object's implementation are hidden.

3. Inheritance Allows classes to use parent classes behaviour & structures.
   - Improves reliability & manageability.
   - Allows code reusability

4. Polymorphism
   Different implementations can be hidden behind a common interface.



Procedural

functional

Object – oriented

Logical

Declarative (Rule based)

Imperative (Command driven)

1.

2.

Advantages

1. The system solves the problem, so the programming steps themselves are kept to a minimum.
2. Proving the validity of a given program is simple.

③ Functional

↳ It views all subprograms as functions in the mathematical sense - informally, they take in arguments and return a single solution.

Functional languages permit functional solutions to problems by permitting a programmer to treat functions as first class objects.

Advantages

1. The high level of abstraction, especially when functions are used, supresses many of the details of programming & thus removes the possibility of committing many classes of error.
2. Evaluation order independence (lack of dependence on assignment opr.) It makes functional oriented languages good candidates for programming massively parallel computers.

Disadvantages

1. less efficiency
2. Problems involving many variables or a lot of sequential activity are sometimes easier to handle imperatively or with object-oriented programming.

Features of Java

Lecture 2

What & Why?

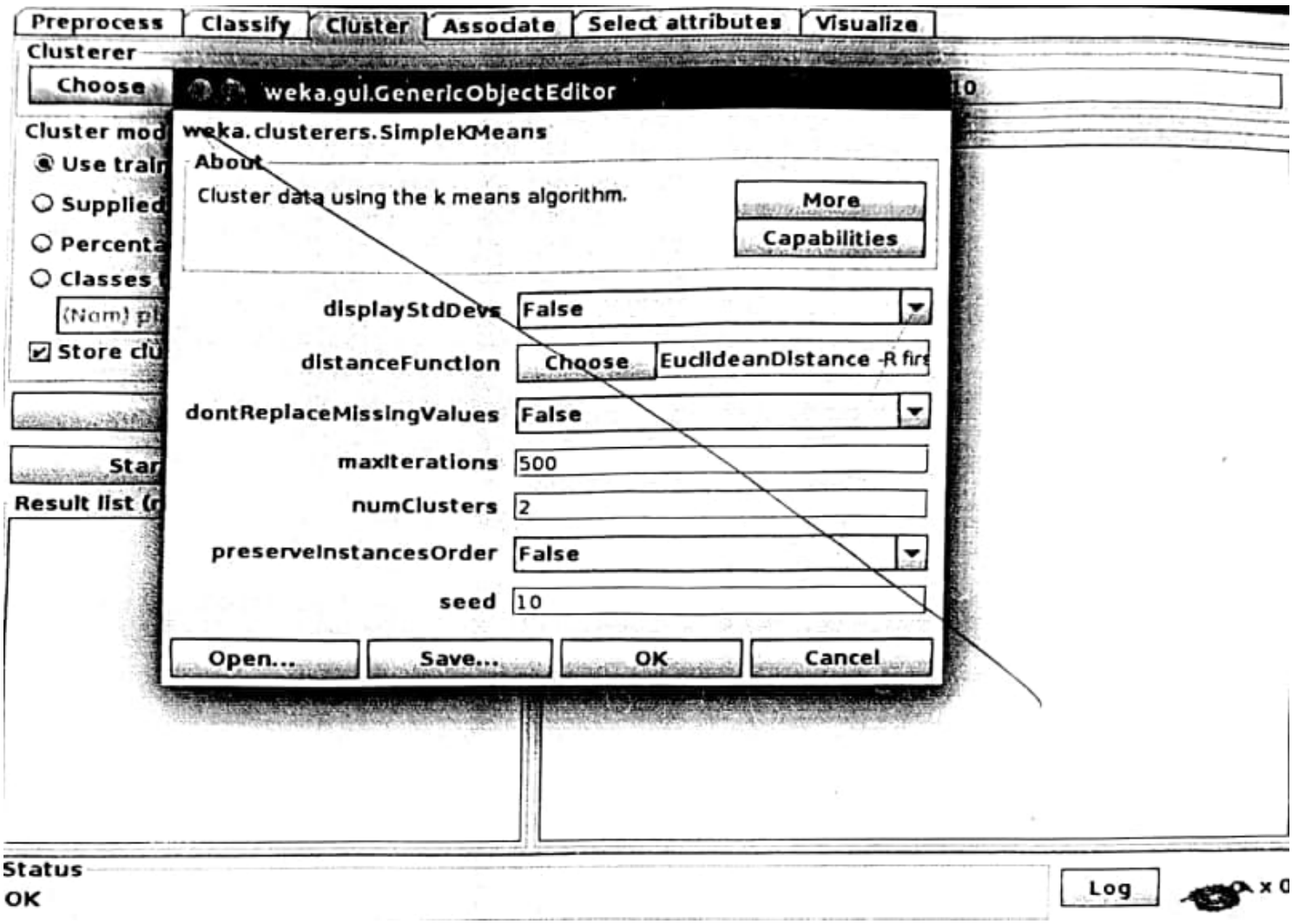
Java is a programming language developed by the Green Project Team of the Sun Microsystems, headed by James Gosling, in the late 1990's.

Most prominent reason why Java became so successful is that it was the first programming language which was not chained to one OS.
Java is designed to be platform independent.
Never before did any lang. poses such a remarkable feature.

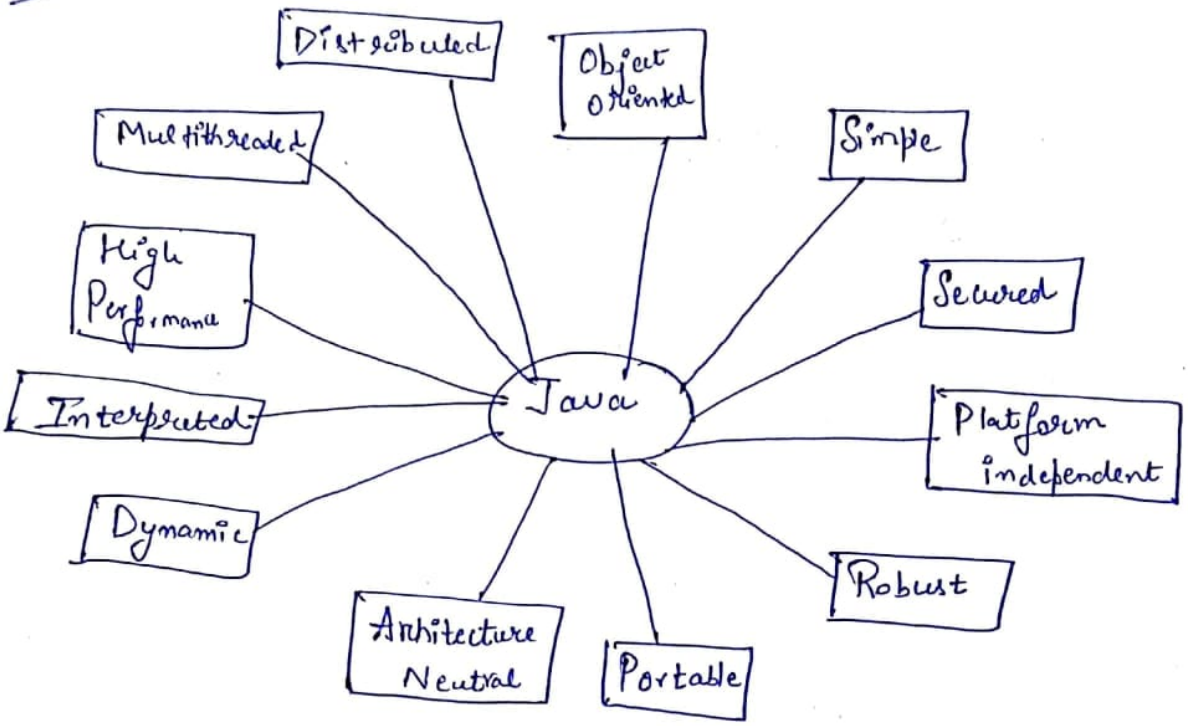
Java is a programming lang. & a platform.

Java is a high level, robust, secured, & OOP lang.

Platform – Any h/w or s/w environment in which a program runs, is known as a platform.
Since Java has its own runtime environment (JRE) and API, it is called platform.



Features of Java



Example