OOP :- OOP stands for Object-Oriented Programming As the name suggests uses objects in Programming Object-Oriented programming aims to implement real world entities like inheritance, Polymorphism etc in Programming.

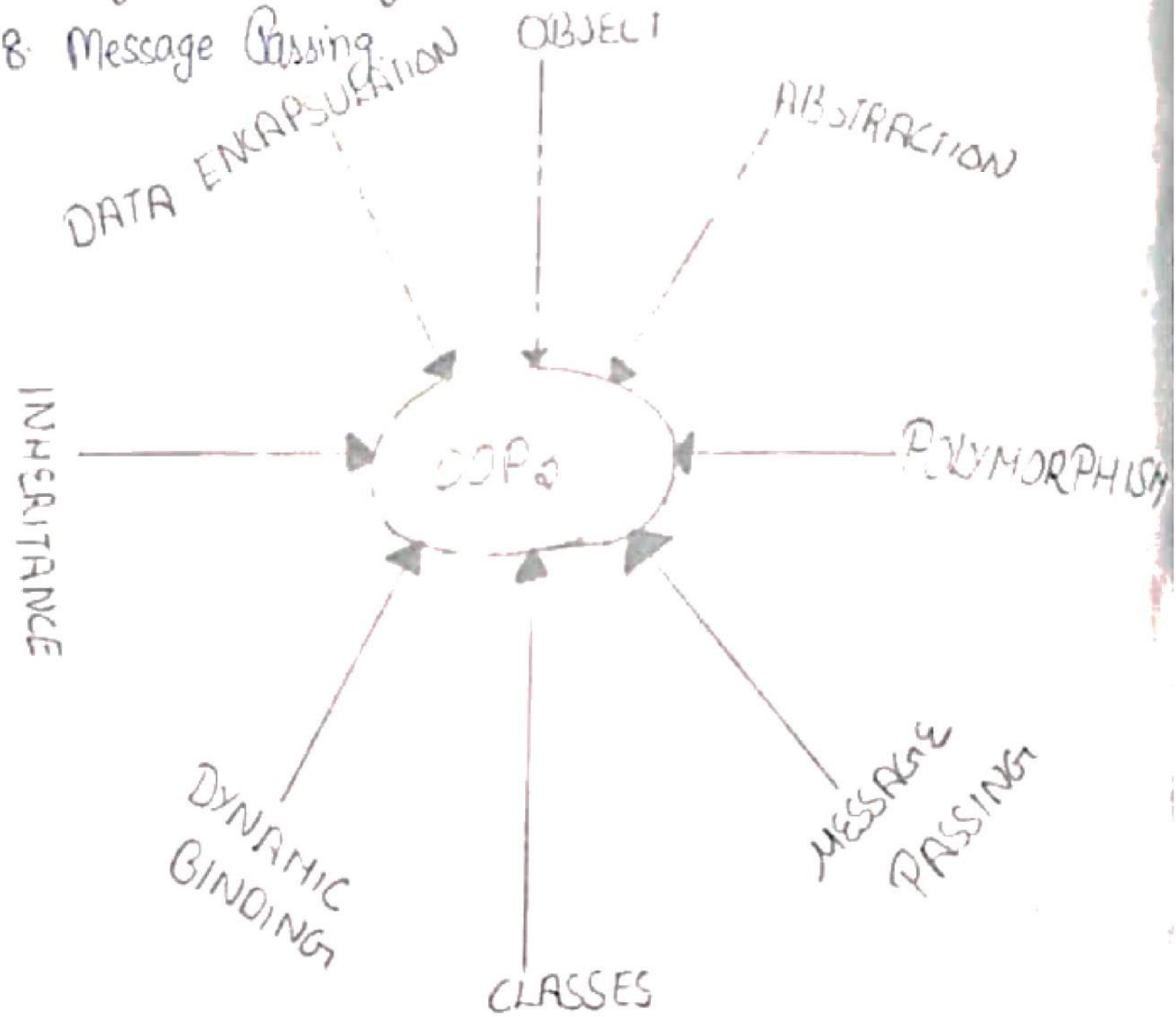
The Primary Purpose of C++ Programming was to add object orientation to the C Programming language which is in itself one of the most powerful Programming languages.

Features of Object Oriented Programming :-

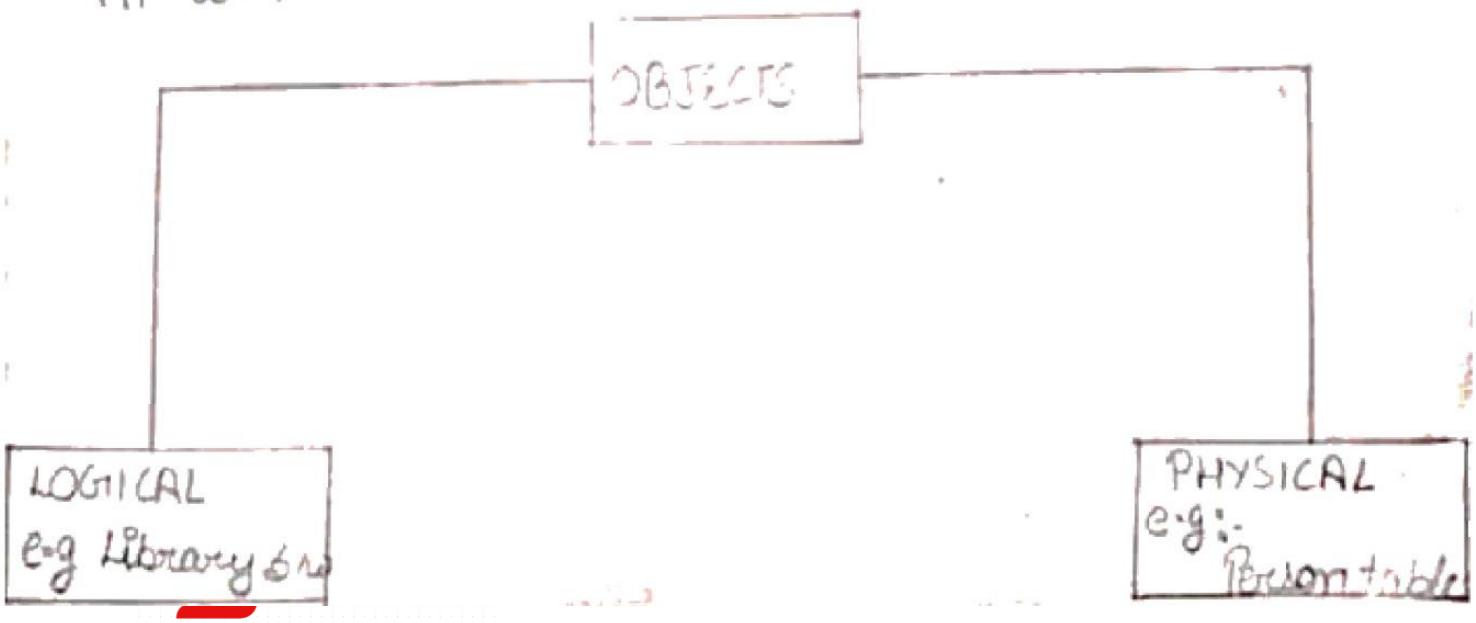
⇒ Programs are divided into units called "Objects".
⇒ Emphasis on data rather procedure.
⇒ Objects used to communicate with each other through functions.
⇒ New functionality can be easily added whenever necessary.
⇒ follows botlow-up design in Program design.
⇒ Data is hidden and cannot be accessed by

Basic Concepts or elements or Components of OOP :-

1. Object
2. Classes
3. Inheritance
4. Polymorphism
5. Data Encapsulation
6. Abstraction
7. Dynamic Binding.
8. Message Passing



* Object means a real world entity such as pen, chair, table etc.
* Any entity that has state and behaviour is known as an object.
* It can be physical and logical.
* It has physical Extension and occupy memory you can create many objects from a class once you have defined a class
* for example :- chair, pen, table, keyboard, bike etc.
* Objects are further subdivided into physical Objects and logical Objects
* Physical Objects, has physical attributes and can be used for real life purpose. for example :- a person, a place, a bank account etc.
* logical Objects are rarely used for data processing in Computer System.



2 Classes

* Collection of Objects is called class. It has no Physical Collection and Occupy no memory.
* A class is way to bind the data and its associated functions together.
* Once a class defined, we can create any number of objects belonging to that class.
* A Class can defined unlimited number of Objects.
* for example:- apple, mango and Orange members of class fruit.
* It can be represented as:-

```
class name obj1, obj2 .... objn;
```

→ here Obj1, obj2 ... objn are different objects of similar class.

⇒ If fruit has been defines as a class, then the statement

```
Fruit Mango;
```

Fruit ← Class
Mango ← object

Create an object Mango, belonging to class fruit.

3 Inheritance

To drive a new class from existing one is called Inheritance.

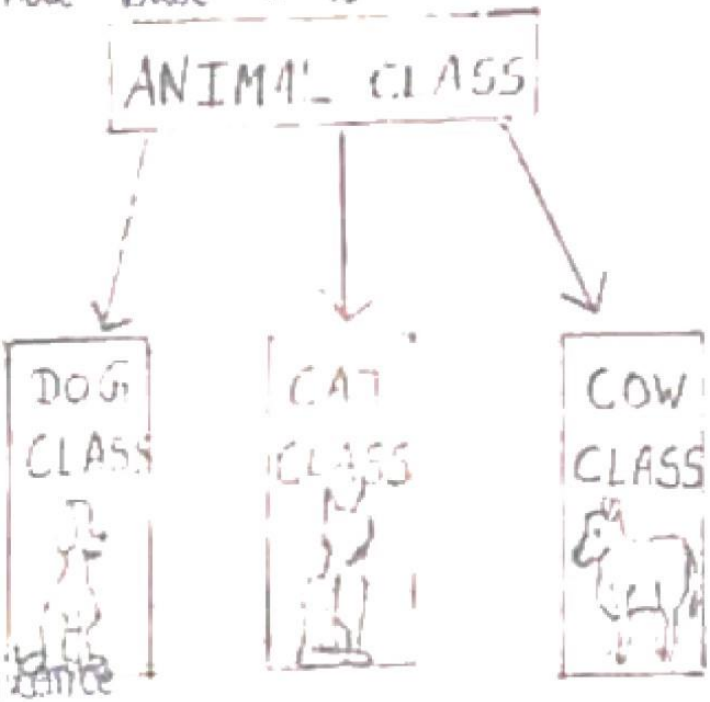
→ New class is also known as sub class or drive class.

→ Old class is also known as super class or base class

It is in hierarical Order.

Inheritance is one of the most important features of Object- Oriented Programming.

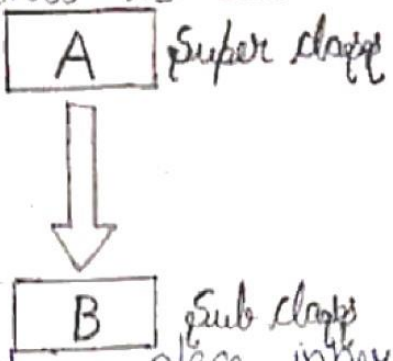
for example Dog, Cat, Cow can be derived Class of Animal Base class



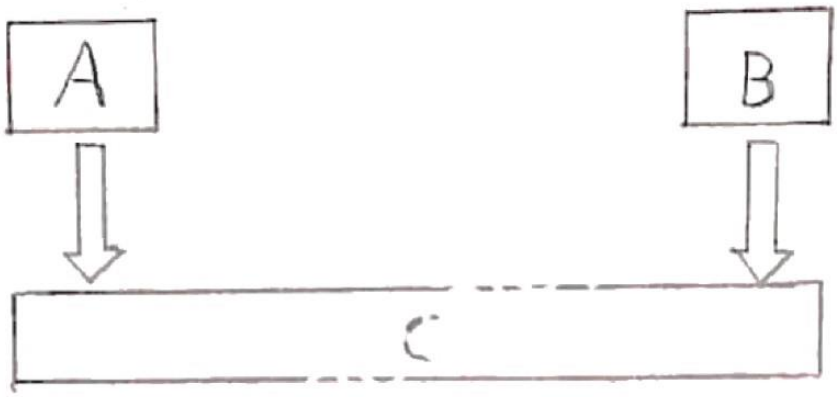
Types of Inheritance

1 Single Inheritance
2. Multiple Inheritance
3. Multilevel Inheritance.
4. Hybrid Inheritance.
5. Hierarchical Inheritance.

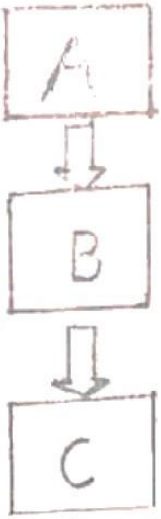
① Single Inheritance:- To drive a class from an existing base class is called Single Inheritance.



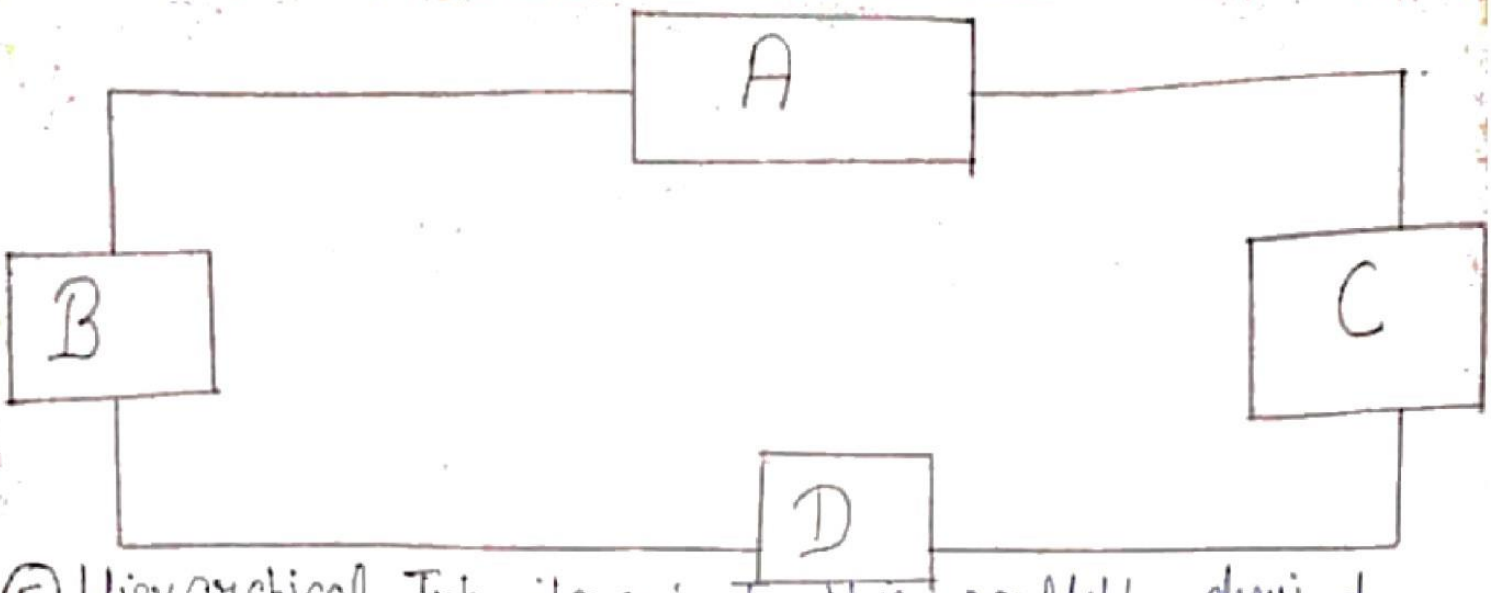
② Multiple Inheritance:- To drive class inheriting from multiple base class is called multiple Inheritance.



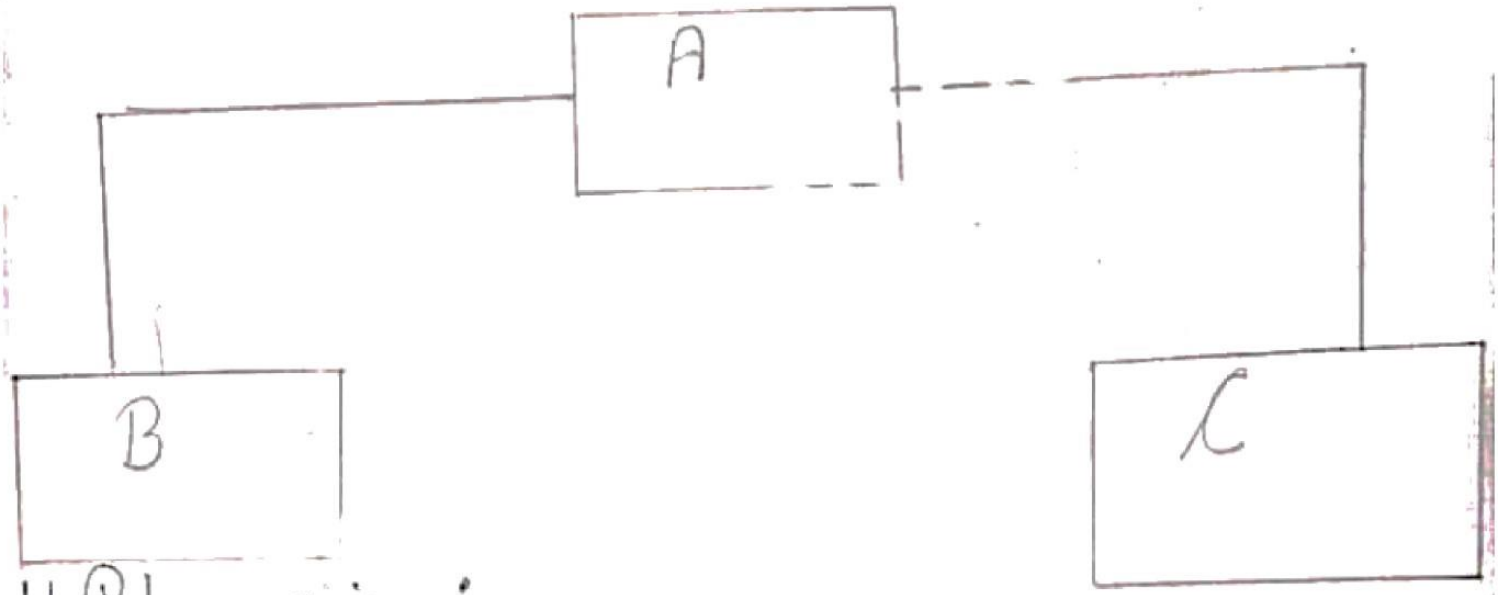
③ Multilevel Inheritance:- To derived class inheriting from a class that inherits from another class is called multilevel Inheritance.



④ Hybrid inheritance:- Hybrid inheritance is Combination of Hierarchical and multilevel Inheritance.

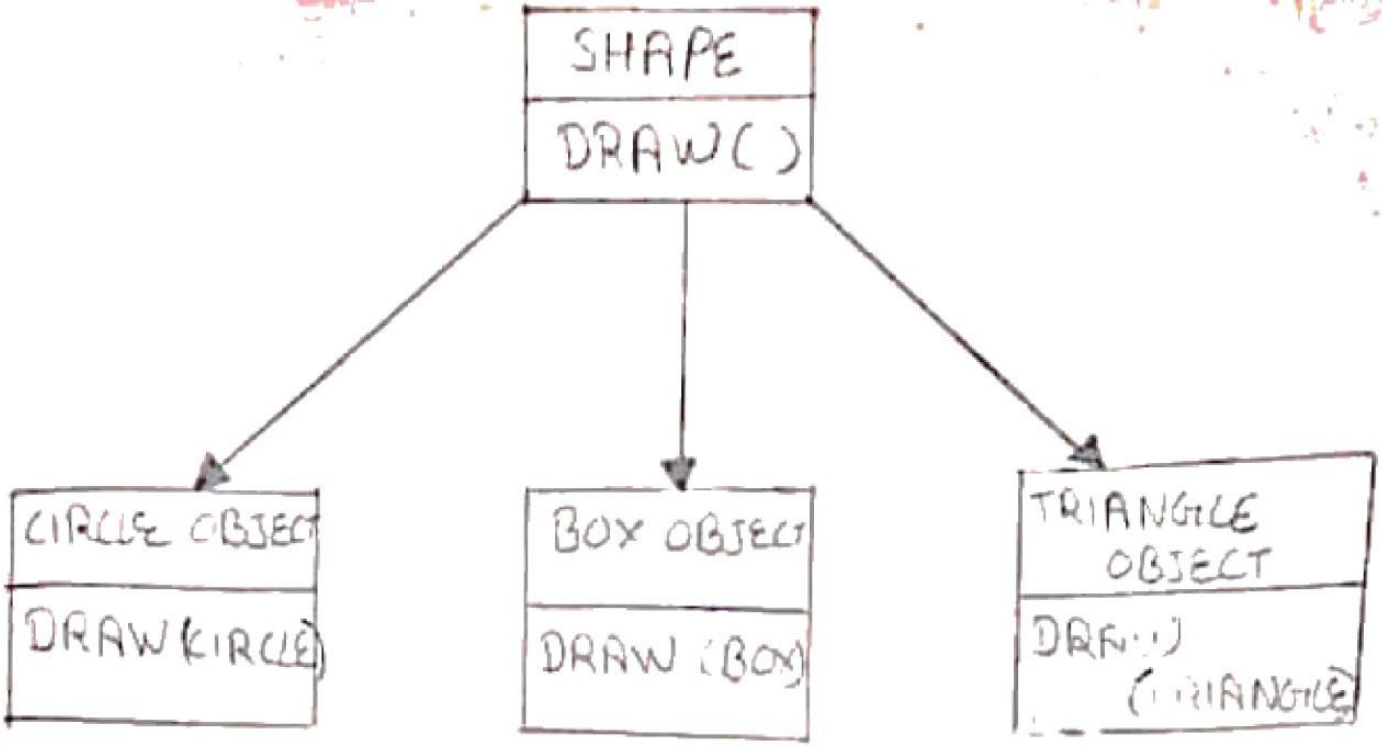


(5) Hierarchical Inheritance :- In this, multiple derived classes inherits from a single base class.



4 Polymorphism :-

* Polymorphism is another important OOP Concept.
* In Greek term, Poly means 'many' and 'morphism' means 'forms'. and thus Polymorphism means 'many forms'
* It is ability to take many shapes or forms.
* It is very close to Inheritance.



* Polymorphism has overloading and virtual function classifications

⇒ Virtual function can be simple virtual function and pure virtual function.

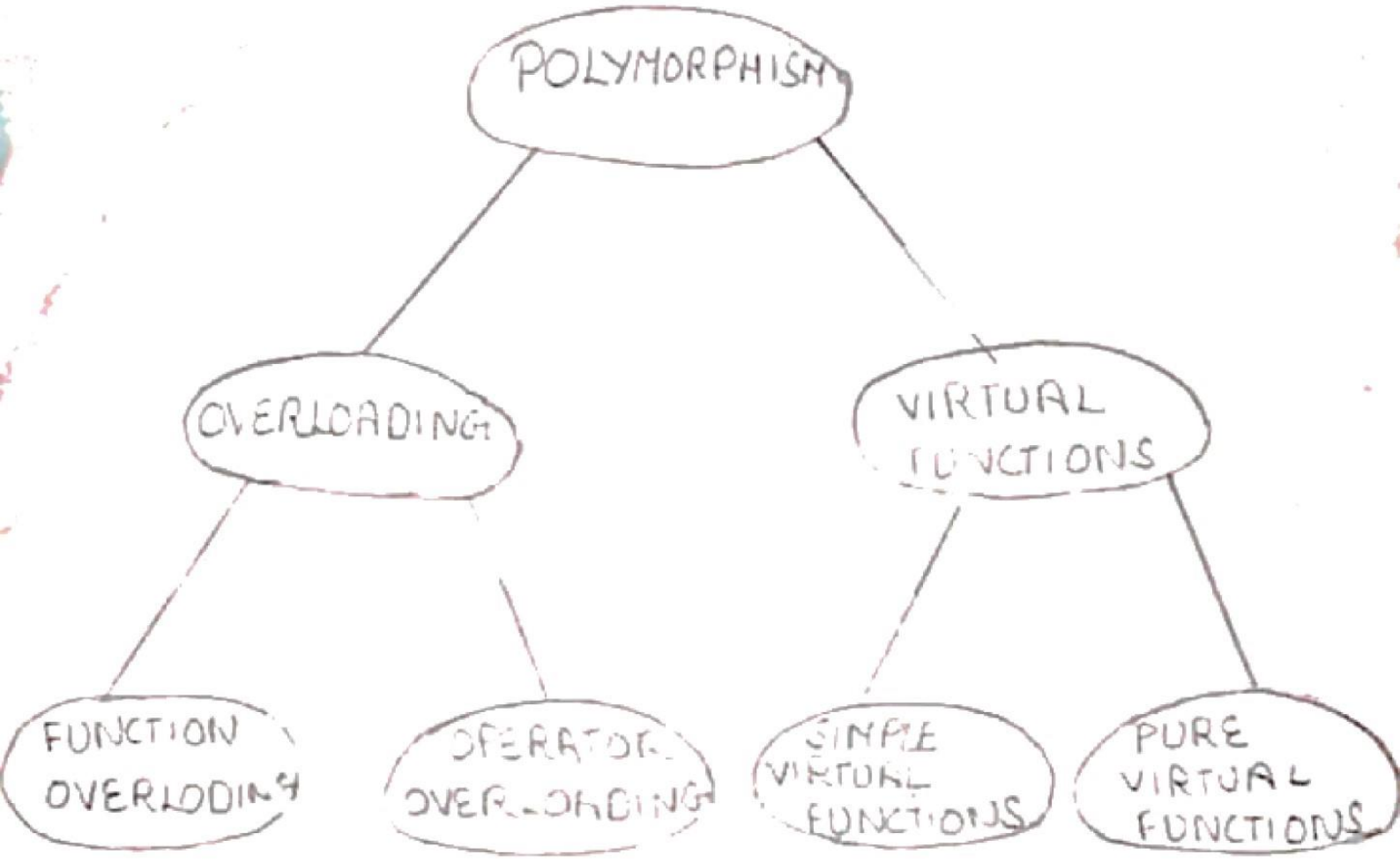
⇒ Overloading is of two types:- function overloading and Operator Overloading (like unary operator overloading, arithmetic operator overloading, assignment operator overloading and binary operator overloading.

* for Example:- if '+' is an operator, then it can be used for two different forms.

⇒ firstly it can concatenate two different Strings "Palvi" and "Arora" as "Palvi" + "Arora" and create a string "Palvi Arora".

⇒ Second, "+" operator can be used to find the sum of two different integer values

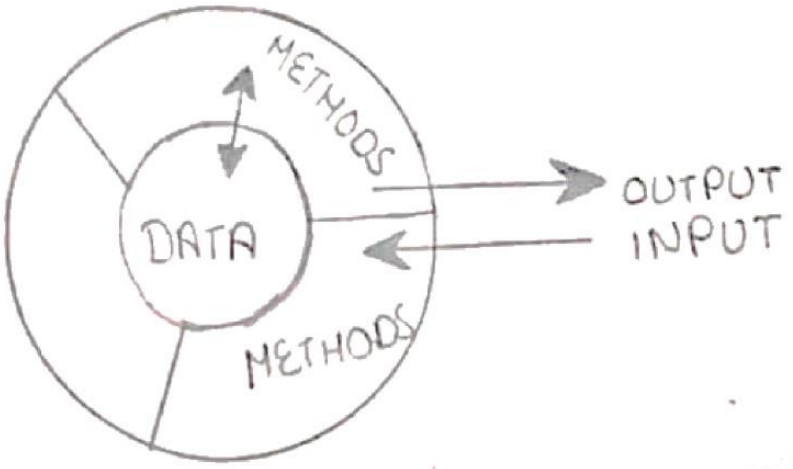
4+3 result is 7.



The keyword "Virtual" is used to polymorphism Concept in C++.

5. Encapsulation (Data Encapsulation):-

* The Wrapping up of data and functions into a single unit known as encapsulation.
* The data is not accessible to the outside world and only those functions which are wrapped in the class can access it.
* These functions provide the interface between the Object's data and Program.
* Encapsulation is used to apply the Principle of data hiding.
* It is also known as data hiding or Information hiding.
* With encapsulation, all of methods in an object can get the data, and Protected from outside access.



6. Abstraction and Data Abstraction

* Abstraction is one of the Powerful Concept of OOPs.
It is very close to Encapsulation.

* Abstraction refers to the act of representing essential features without including the background details and explanations.
⇒ It means displaying only essential information and hiding the details.

* For Example: Phone call, we don't know the internal processing.

* There are two types of Abstraction:-
1. Procedural Abstraction.
2. Data Abstraction.

7. Binding:- Binding means linking. There are two types of Binding.

(i) Static Binding.
(ii) Dynamic Binding.

(i) Static Binding:- The Code associated with a given procedure is known

It is also known as early Binding

⇒ Static Binding is less reliable.

⇒ In this, Program execution is fast

⇒ Example: function Overloading, Operator Overloading

⇒ Events Occurs at Compile time are "Static Binding".

(ii) Dynamic Binding:-

In dynamic Binding, the code associate with a given procedure is known at run time

⇒ It is also known as Late binding.

⇒ Program execution is little slow.

⇒ Dynamic Binding is more reliable.

⇒ Example: Virtual function.

(8) Message Passing:-

Message Passing is nothing but sending and receiving of information by the objects same as people exchange information.

* It is a method by which an objects sends data to another object or requests other objects to invoke (call) method. This is also known as Interfacing.