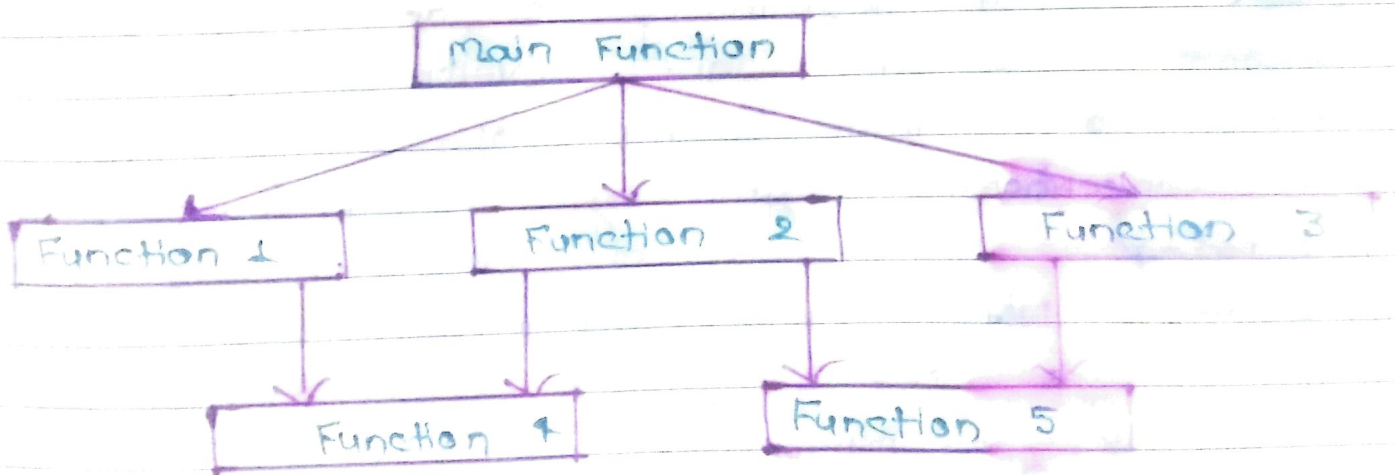


M	T	W	T	F	S	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29				

Procedure Oriented Programming

High level language such as COBOL, FORTRAN, C etc are known as procedure oriented programming. Procedure oriented programming approach is an approach in which a problem is viewed as a sequence of things to be done such as reading, calculating or manipulating and printing. We can use number of functions to accomplish this task. In procedure oriented programming, the primary focus is function. The structure for procedure oriented programming is given below.



Characteristics of Procedure Oriented Programming

> In procedure oriented programming stress is given to doing thing (algorithm)

M	T	W	T	F	S	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

> Program is divided into smaller program known as function.

> Most of the function shares global data.

> Data openly moves around the system from one function to another.

> It follows top-down approach.

> Function transform data from one function to another.

Introduction To Object Oriented Programming

Object Oriented programming is an approach to software organization and development that attempts to eliminate some of the drawbacks of conventional programming method by incorporating the best of structure programming with several powerful new concepts. It is a new way of organization and designing programs and has nothing to do with any particular organization also all languages are not suitable to implement OOP concept easily.

	M	T	W	T	F	S	S
			1	2	3	4	5
FEB	6	7	8	9	10	11	12
2012	13	14	15	16	17	18	19
	20	21	22	23	24	25	26
	27	28	29				

JANUARY

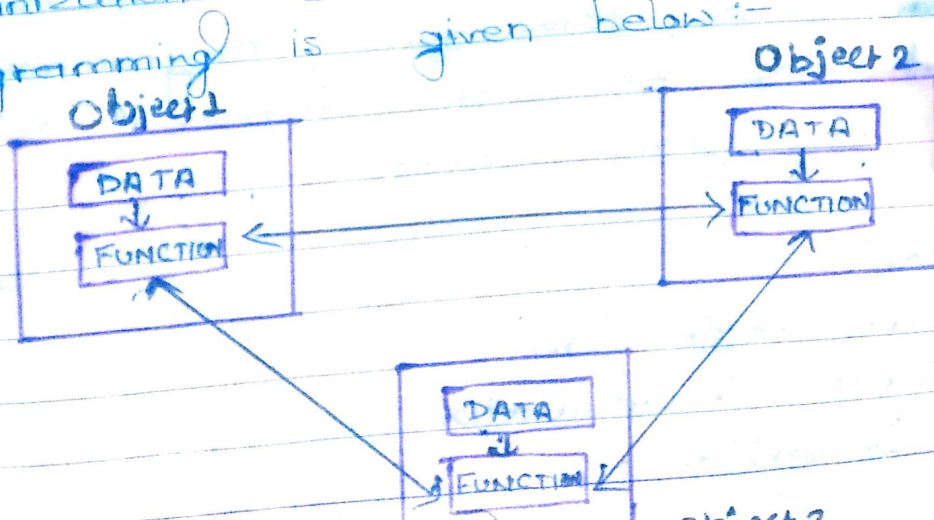
TUESDAY
003-363 Week 01

03

of modularizing problem by decomposing a problem into number of entity called 'Data' and 'Function' around the objects. The data of the object can be accessed only by the function associated with that object. However, the function of an object can be accessed by other object.

OR

In object oriented programming data are critical element in the program development. Object Oriented programming concept doesn't allow to flow freely around the system. It ties data more closely to the function that operates on it and protect it from accidental modification from outside function. Object oriented programming allows decomposition of a problem into number of entities called object and then build data and function around these object. Its organization of data function in object oriented programming is given below :-



M	T	W	T	F	S	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Characteristics Of Object Oriented Programming

- 10 > Stress is given on data rather than procedure.
- > Programs are divided into objects.
- 11 > Data is hidden and can not be accessed by external function.
- 12 > Object may communicate with each other through function.
- 1 > New data and function can be easily added whenever necessary.
- 2 > It follows bottom-up approach.

Important Features Of OOPs

3 The object oriented programming has been developed to overcome the drawback of procedural programming. The object oriented approach based on certain concept that help it to attain its goals of overcoming the drawbacks of procedural programming. The general concept of object oriented programming is given below -

> class

> object

> Data Abstraction

> Data Encapsulation

> Inheritance

	M	T	W	T	F	S	S
			1	2	3	4	5
FEB	6	7	8	9	10	11	12
2012	13	14	15	16	17	18	19
	20	21	22	23	24	25	26
	27	28	29				

JANUARY

THURSDAY
005-361 Week 01

05

> Polymorphism -

> Dynamic binding

> Message passing

Class:- A class is user defined data type used to represent a template (Format) for several similar type of object. The entire set of data and code of an object can be made a user defined data type with the help of class. Once a class has been defined we can create any no. of object belong to that class. For ex:- mango, Apple, Orange, grapes etc. are objects of the class fruit.

Class fruit

{

char fname[50];

char fcolour[50];

char ftaste[50];

} f₁, f₂, f₃;

Object:- Object is a basic building block for object oriented programming. It corresponds to real life entities. An object is an unit of software consist of :-
* state / Attribute

M	T	W	T	F	S	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

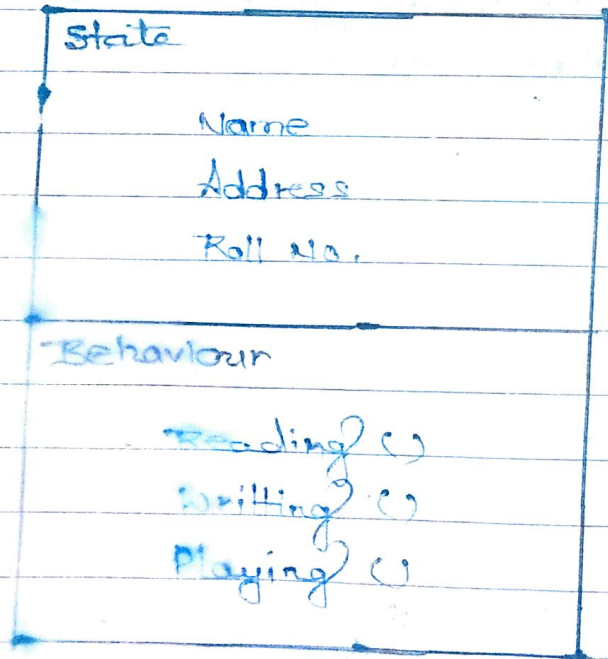
* Behaviour / Function

* Identity

* State :- state or attribute defines the property of an object and current value of each of these properties.

* Behaviour / Function :- Behaviour defines the response of an object reacts in terms of its state changes and message passing.

* Identity :- Identity defines an unique name



Data Abstraction :- Abstraction refers to an act of representing essential features without including the background details or explanation. For ex:- we are

	S	M	T	W	T	F	S	S
				1	2	3	4	5
FEB	6	7	8	9	10	11	12	
2012	13	14	15	16	17	18	19	
	20	21	22	23	24	25	26	
	27	28	29					

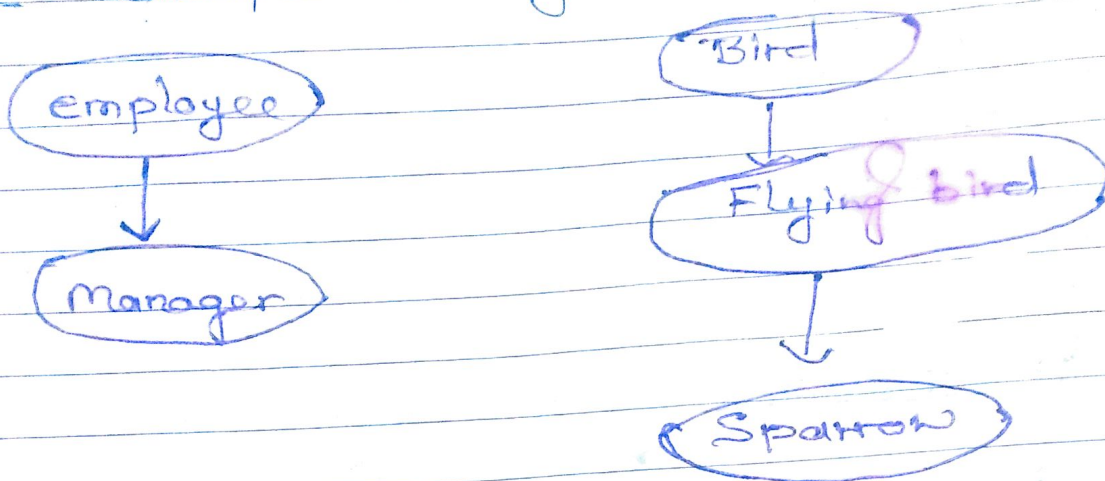
driving a car. We only know the essential features to drive, such as - gear handling, steering handling, uses of clutch, accelerator, breaks etc. But while driving we do not know the interval details of car like waiting or motor working.

Data Encapsulation :- Encapsulation is the way of combining both data and function that operates on the data under a single unit or we can say that the grouping of data and function into a single unit (class) is known as encapsulation.

Inheritance :- Inheritance is the process by which an object of one class acquires properties of another class. For example, the bird is a part of class 'flying' and the manager is the part of class 'employee'. The class manager acquires the characteristics of class employee. Hence, manager is a derived class and employee is the base class. The main principle behind inheritance is that a derived class shares common characteristics with the base class from which it derives.

The graphical representation of

above example is given below:-



In object oriented programming the concept of inheritance is the idea of reuseability. of inheritance This means that we can add additional features on existing class without modifying it. This is possible by deriving a new class from existing one. The new class will have combined features of both the classes i.e. 'Base' and 'Derived'.

Classification Of Inheritance

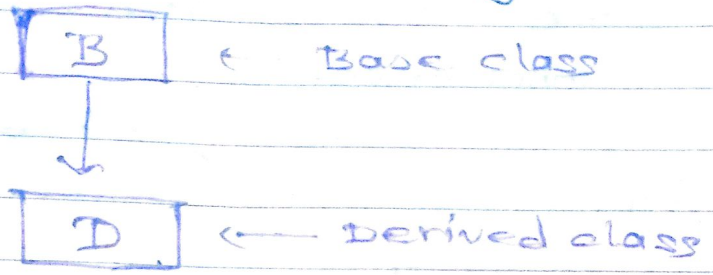
Depending upon how the classes are derived from the base class inheritance can be classified into the following :-

(i) Single inheritance:-

When a class is derived from the base class is known as

	T	W	T	F	S	S
		1	2	3	4	5
	6	7	8	9	10	11
FEB	12	13	14	15	16	17
2012	18	19	20	21	22	23
	24	25	26	27	28	29

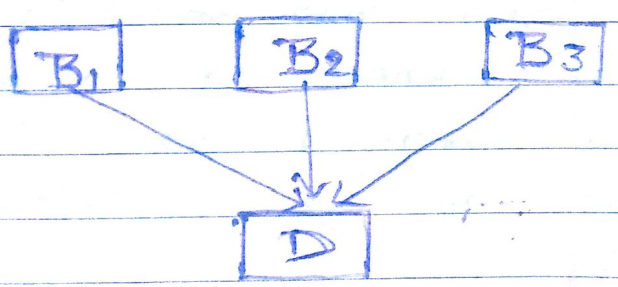
Single inheritance. The graphical representation of single inheritance is given below:-



In the above figure, 'B' represents the 'Base class' and 'D' represents the 'Derived class'.

(ii) Multiple Inheritance :- Inheritance in which a derived class has several base classes is known as multiple inheritance.

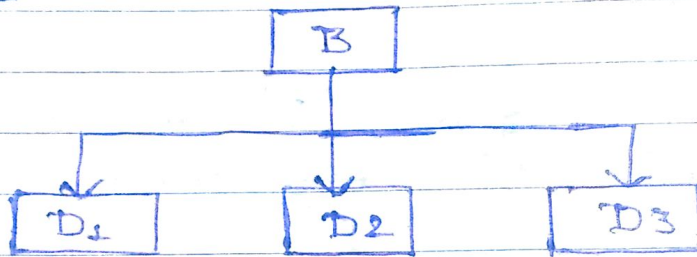
The graphical representation of multiple inheritance is given below:-



In the above figure B₁, B₂ and B₃ are 'Base classes' and D is 'Derived class' which contains the property of B₁, B₂ and B₃.

ii) Hierarchical Inheritance :- When one class is inherited by more than one classes then it is called hierarchical inheritance.

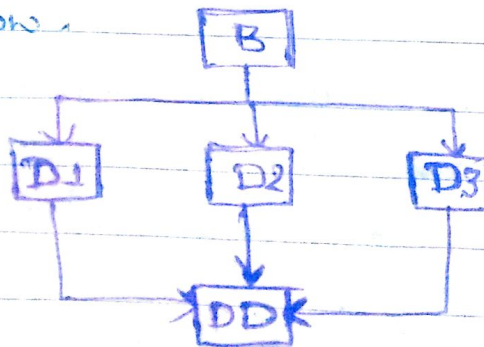
The graphical representation of hierarchical inheritance is given below :-



In the above example, B is the 'Base class' and D1, D2 and D3 are derived classes. The properties of Base class 'B' can be shared by D1, D2 and D3 classes.

iii) Hybrid Inheritance :-

Hybrid inheritance is a method of inheritance where, a class is derived from several derived classes. The graphical representation of hybrid inheritance is given below.



	M	T	W	T	F	S	S
			1	2	3	4	5
FEB	6	7	8	9	10	11	12
2012	13	14	15	16	17	18	19
	20	21	22	23	24	25	26
	27	28	29				

(v) Multi-Level Inheritance :- In a multilevel inheritance a class is derived from another derived class. The graphical representation of multi-level inheritance is given below:-

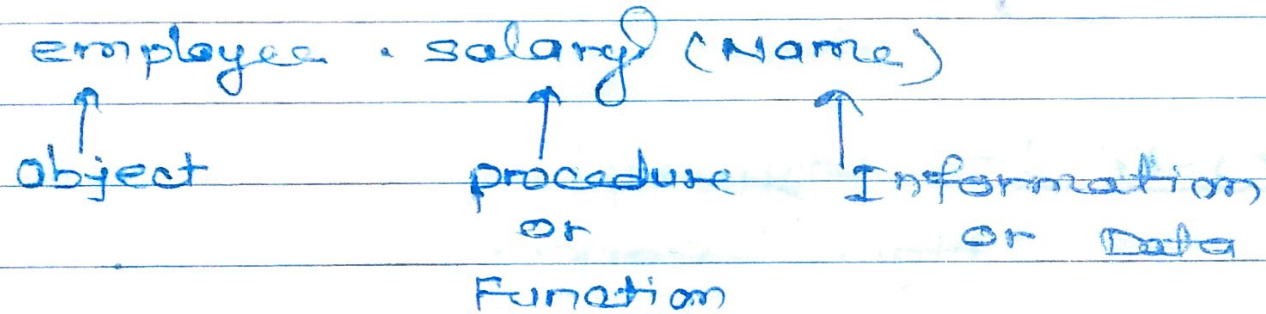


Polymorphism :- Polymorphism is the important concept of object oriented programming. Polymorphism is a Greek term means "the ability to take more than one form". It is the ability for a message or data to be placed in more than one form. It is the property by which several different object response in different ways.

Dynamic Binding :- Binding refers to linking a called procedure to the code to be executed in response to the call. Dynamic binding means that the code associated with a given procedure is not known until the time of call at run time. It is associated with polymorphism and

object and information to be sent.

Ex: -



Advantages Of Object Oriented Programming

- > It models real world well.
- > Object oriented programs are easy to under-

	M	T	W	T	F	S	S
			1	2	3	4	5
FEB	6	7	8	9	10	11	12
2012	13	14	15	16	17	18	19
	20	21	22	23	24	25	26
	27	28	29				

> Object Oriented programs are easy to test, maintain and manage.

Disadvantage Of Object Oriented Programming

> The object oriented programming design is tricky (logic)

> One need to prepare planning and prepare design for object oriented programming.

> To program with OOP programmers need prepare skill such as :- design skill, thinking in term of object.

Difference between C and C++

C++	C
(i) In C++ file name have .cpp extension.	(i) In C file name have .c extension.
(ii) In C++ identifiers can have any no. of character.	(ii) In C only 31 characters are significant.
(iii) Default value for the parameters in the function prototypes possible in C++	(iii) In C this facility is not available.