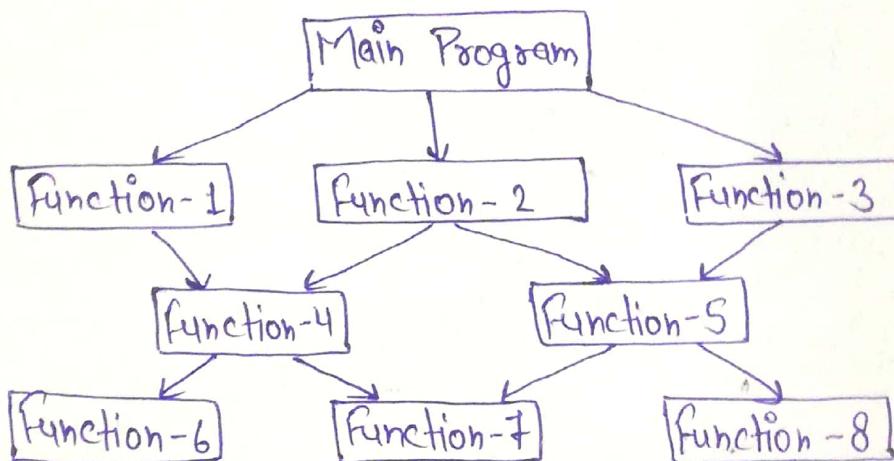


LECTURE - I# Procedure Oriented Programming Language :-

Conventional programming using high level language such as cobol, fortran and c is commonly known as procedure Oriented programming language. In this language the problem is viewed as a sequence of things to be done such as reading calculating and printing. A number of functions are return to achieve these tasks.

The primary focus is on functions. A typical program structure for procedure Oriented programming language is given in figure below.

* Characteristics of procedure Oriented programming language :-

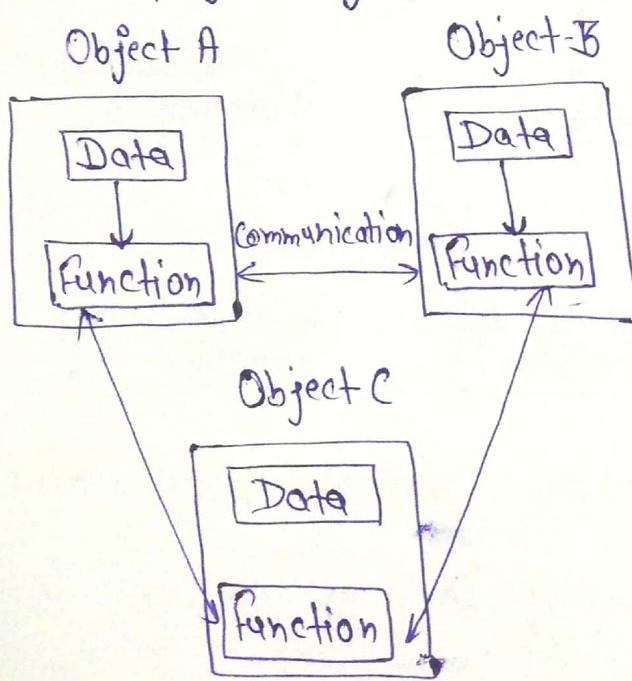
- i Procedure Oriented programming language emphasis is on doing things (algorithms).
- ii Large programs are divided into smaller programs which is known as functions.
- iii Most of the functions in Procedure Oriented programming language share the global data.
- iv data move openly around the system from function to function,
- v function transform data from one form to another form.

V^o Procedure Oriented Programming language focus on Top-down approach in program design.

Object Oriented programming Language :-

The major motivating factor in the invention of Object Oriented approach is to remove some disadvantages of Procedure Oriented approach. In Object Oriented Programming our main focus is towards data.

Object Oriented programming allow decomposition of a problem into a number of entities which is known as objects and then builds data and functions around these objects. The organization of data and functions in Object Oriented programming is shown in figure.



The data of an object can be accessed only by the functions associated with that object. However the functions of one object can access the functions of other objects.

* Characteristics of Object Oriented programming language :-

1. Object Oriented approach emphasis is on data rather than procedure.
2. Programs are divided into objects.
3. Data structures are designed such that they characterize the objects.
4. Functions that operate on the data of an object are tied together in the Data Structure.
5. Data is hidden and can't be accessed by external functions.
6. Objects may communicate with each other with the help of functions.
7. New functions and Data can be easily added whenever required.
8. Object Oriented programming approach follow bottom-up approach in program design.

Difference between Procedure Oriented programming language and object Oriented programming language.

POP language

1. POP language emphasis is on doing things (Algorithm).
2. Program divide into smaller program called function.
3. Function transform data from one form to another form.
4. POP language is focus on Top-down approach in program design.

OOP language

1. Object Oriented approach emphasis is on data rather than procedure.
2. Programs are divided into objects.
3. Data is hidden and can't be accessed by external functions.
4. OOP approach follows bottom-up approach in program design.

LECTURE - 2

Features / Characteristics / basic concept of Object Oriented language :-

1. Object :- Object is a real world entity which have ability to access variables and methods of a class.

2. Class :- Class is a userdefined datatype which contains variables and methods.

3. Inheritance :- Inheritance is also known as re-usability of code the term inheritance may be defined as the process to extend features of parent class into child class. Sometimes instead of parent class we use superclass or base class and instead of child class we also use subclass or derived class.

There are 5 types.

- i. Single level.
- ii. Multilevel
- iii. Hierarchical
- iv. Multiple
- v. Hybrid.

= Polymorphism :- Polymorphism means more than one form. In a programming environment when same method behaves differently then it is known as polymorphism.

Data Encapsulation :- The term Data Encapsulation is the hiding of data members (variables) and member functions into a single unit is known as Data Encapsulation.

6. Data Abstraction :-

Data Abstraction is the process to hide unnecessary information and only representing the necessary information to the user is known as Data Abstraction.

7. Information Hiding :-

Information Hiding is the process in which we represent essential features without including the background details or explanations or implementation.

8. Dynamic Binding :-

It is the process to transfer data between different classes or modules at the run time. is known as Dynamic Binding.

9. Message Passing :-

In OOP, objects communicate with each other using messages. When objects communicate, information is passed back and forth between the objects. A message generally consists of the object name, method name and actual data that is to be sent to another object.

Benefits of Object Oriented Programming :-

OOP offers several benefits to programmer as well as users.

Object Oriented contributes to the solution of many problems associated with the development and quality of software products. The new technology offers greater programmer productivity better quality software and lesser maintenance cost.

The Principle Advantages are :-

1. With the help of inheritance we can eliminate redundant code and extend the use of existing classes.
2. We can build programs from the standard working modules that communicate with one another rather than having to start writing the code from scratch. This process leads to saving of development time and higher productivity.
3. The principle of data hiding helps the programmer to build secure programs that can't be modified by code in other parts of the program.
4. It is possible to have multiple instances of an object to co-exist without any interference.
5. It is possible to map objects in the problem domain to those in the program.
6. It is easy to partition the work in a project based on objects.
7. The data centred design approach enables us to capture more detail of a model in implementable form.
8. Object Oriented system can be easily upgraded from small to large systems.
9. Message passing techniques for communication between objects make the interface description with external systems much simpler.
10. Software complexity can be easily managed.

LECTURE - 3

①

Applications of OOP :-

- i Real time systems.
- ii Simulation and modeling.
- iii Object Oriented databases.
- iv Hypertext and Hypermedia.
- v AI and Expert systems.
- vi Neural networks and parallel programming.
- vii Decision support and office automation system.
- viii CIM / CAD / CAM systems.

Object Oriented Analysis :-

Analysis is the process of transforming a problem definition from a fuzzy set of facts and myths into a coherent statement of system's requirement.

Tools used for Object Oriented Analysis are :-

Examination of existing system document, Interview, Questionaries and Observations.

Analysis is a creative activity that involve understanding the problem, its associated constraint and methods of overcoming those constraints. This is an iterative process and goes until problem is completely understood.

Common problems in Analysis are :-

- i Fuzzy and ambiguous requirements.
- ii Incomplete requirements.
- iii Unnecessary features.

To understand the user requirements we need to find out how they use the system.

- 1. This can be achieved by creating use-cases.
- 2. Try to understand the unexpected inputs and desired responses.
- 3. Preparation of a prototype usually can help you better understand how the system will be used and therefore it is a valuable tool during business object analysis.
- 4. Fully understanding a problem and defining how to implement it and it may require several transactions and iterations.

Object Oriented design :-

An object is an instance of a class which contains meaningful data. It concepts that occupy memory space at runtime according to the class definition.

Characteristics of Object Oriented design :-

- 1. Objects are abstractions of real world or system entities and manage themselves.
- 2. Objects are independent and they contain encapsulating state and representation information.
- 3. System function expressed in term of object services.
- 4. Shared data areas are eliminated (Objects communicate by message passing).
- 5. Objects may be distributed.
- 6. Objects may execute sequentially or in parallel.

LECTURE - 4

* Advantages of Object Oriented design :-

- i Object Oriented design is easier to maintain or we can say that objects may be understood as stand alone entities.
- ii Objects are appropriate reusable components.
- iii For some systems there may be an obvious mapping from real world entities to system objects.

Combining Three Models :-

- ① Object Model.
- ② Dynamic Model.
- ③ Functional Model.

a Object Model :- An object Model consist of the following important features.

a. Object Reference :- Objects can be accessed via object references to invoke a method in an object the object reference and method name are given together with any arguments.

b. Interface :- An interface provides a definition of the signature of a set of methods without specifying their implementation. An object will provide a particular interface if its class contains code that implement the methods of that interface. An interface also define types that can be used to declare the types of variables and parameters and return values of methods.

c: Actions :- Action is an object Oriented programming is initiated by an object invoking a method in another object. The invocation can include additional information needed to carry out the method the receiver executes the appropriate method and then returns control to the invoking object.

d: Exception :- Programs can encounter various errors and unexpected conditions, due to which program is not working properly. During the program execution some methods discover different kind of problem at runtime. Exception provide a clean way to deal with error conditions without complicating the code. A block of code may be defined to throw an exception whenever particular unexpected conditions or errors occur. This means that control passes to another block of code that catches the exception.

Garbage Collection :- It is necessary to provide the means of free the space occupied by objects when they are no longer needed.

for Ex :-

Java can delete automatically when an object is no longer accessible. Recover the space and make it available for allocation to other objects. This process is known as Garbage Collection.

2. Dynamic Model :-

A dynamic model is used to express and model the behaviour of the system over the time it include activity diagrams, State chart diagram, sequence diagram and extensions including business process modelling.

- 1. Activity Diagram.
- 2. State Chart diagram.
- 3. Sequence diagram.
- 4. Process modelling.

} Ch-2

3. Functional model :-

A function model or functional model in system engineering and software engineering is a structural representation of the functions (activities, actions, processes and operations) within the modeled system or subject area.

A functional model is also known as activity model or process model which is a graphical representation of an enterprise's function within a defined scope.

The purpose of the function model is to describe the functions and process, with assist with discovery of information needs help identify opportunity and establish a basic for determining product and services cost.

JSD X

LECTURE - 5

JSD (Jackson structured development) :-

JSD was first presented by Jackson in 1982. JSD is continuously growing and a few new features were introduced in this method these are described in 1989 by John Cameron.

* Principle of operations :-

Three basic principles of operation of JSD is that

1. Development must start with describing and modelling the real world rather than specifying or structuring the function performed by system. A system made with JSD method performs the simulation of the real world before any direct attention is paid to function or purpose of the system.
2. An adequate mode of a time ordered world must itself be time ordered. Main aim is to map progress in the real world or progress in the system that model it.
3. The way of implementing the system is based on transforming of specification into efficient set of processes these processes should be designed in such a manner that it would be possible to run them on available software and hardware.

JSD steps :-

- 1 Entity Action Step.
- 2 Initial Model Step.
- 3 Interactive function Step.
- 4 Information functions Step.
- 5 System timing Step.
- 6 System Implementation Step.

Later some steps were combined to create a method with only 3 steps.

- 1 Modelling stage (Analysis) :- It is the combination of entity action steps and entity structure step.
- 2 Network Stage (Design) :- It is combination of initial model step function step and system timing step.
- 3 Implementation Stage (Realization) :- It is a system implementation step.

1. Modelling Stage :-

In the modelling stage designer creates a collection of entities structure diagrams and identifies the entities in the system. the action they perform, the time ordering of the action in the life of the entities and the attributes of the actions and entities. Purpose of these diagram is to create a full description of the aspects of the system and the organisation.

2. Network Stage :-

A model of the system is represented with the help of system specification diagram (SSD) which is also known as network diagram. Network diagram represents and it also displays how these processes are communicate with each other. This network diagram is very similar to ER diagrams.

3. Implementation Stage (Realization) :-

The abstract network model of the solution is converted into a physical system & represented with the help of system implementation diagram (SID).

The SID is used to display the system as scheduler process that calls modules that implement the processes. The control concern of implementation step is optimization of the system. It is necessary to reduce the number of processes because it is impossible to provide each process that is contained in specification within its own virtual processor.