Chapter 2 Concepts of Object Oriented Programming

Programming Paradigm: the way in which program is organized. C++ is a multiple paradigm language. It supports both procedural paradigm as well as object oriented paradigm.

Limitations of POP:

1.Data is under valued

2.Adding new data element may require modifications to all/many functions.

3.Creating new data types(Extensibility) is difficult.

4. Provides poor real world modelling.

Object Oriented Programming Concepts

Object:-is an identifiable entity with some characteristics and behaviour.

Class:-is a group of objects that shares common properties and relationships.

```
class class_name
    {
        private: variable declarations;
        function declarations;
        public : :variable declarations;
        function declarations;
    };
```

Object is an instance of a class and class is the **blueprint** of an object.

Message passing means calling member function of an object from another object.

Data Abstraction:-The act of representing essential features without including the background details or explanations.

Encapsulation:-The wrapping up of data and functions into a single unit.

Modularity:-The process of partitioning a big program into several smaller modules.

Inheritance:-The process of deriving a new class(**derived class**) from an existing class(**base class**).

```
class derived_class: AccessSpecifier base_class
    {
```

};

The Access Specifiers are **private, public or protected**.

Different **types of inheritance** are single, multiple, multilevel, hierarchical, and hybrid inheritance.

Polymorphism:-ability to express different forms

1.Compile time(Early Binding) (Static Polymorphism):-Ability of a compiler to bind a function call with function definition during compilation time.

a. Function Overloading: Functions with same name but different signatures can act differently.

b. Operator Overloading: The concept of giving new meaning to an existing C++ operator.

2.Run Time(Late Binding) (Dynamic Polymorphism):- Ability of a compiler to bind a function call with function definition during run time. It uses the concepts of pointers and inheritances.

Advantages of OOP

a. OOP provide a clear modular structure for programs.

b. It is good for defining abstract data types.

c. Implementation details are hidden from other modules and have a clearly defined interface.

d. It is easy to maintain and modify the existing code as new objects can be created without disturbing the existing ones.

e. It can be used to implement real life scenarios.

f. It can define new data type as well as new operations for operators.