**Module: 4.1 Assignment**

**(4.1.2) What is OOP? List OOP concepts**

* OOP stands for Object-Oriented Programming. Procedural programming is about writing procedures or functions that perform operations on the data, while object-oriented programming is about creating objects that contain both data and functions. Object-oriented programming has several advantages over procedural programming:
* OOP is faster and easier to execute
* OOP provides a clear structure for the programs
* OOP helps to keep the C++ code DRY "Don't Repeat Yourself", and makes the code easier to maintain, modify and debug
* OOP makes it possible to create full reusable applications with less code and shorter development time
* It simplifies the software development and maintenance by providing some concepts:
* Object
* Class
* Inheritance
* Polymorphism
* Abstraction
* Encapsulation

### Object Oriented Programming is a paradigm that provides many concepts such as inheritance, data binding, polymorphism etc. The programming paradigm where everything is represented as an object is known as truly object-oriented programming language. Smalltalk is considered as the first truly object-oriented programming language.

### Object

Any entity that has state and behavior is known as an object. For example: chair, pen, table, keyboard, bike etc. It can be physical and logical.

Class

Collection of objects is called class. It is a logical entity.

A Class in C++ is the foundational element that leads to Object-Oriented programming. A class instance must be created in order to access and use the user-defined data type's data members and member functions. An object's class acts as its blueprint. Take the class of cars as an example. Even if different names and brands may be used for different cars, all of them will have some characteristics in common, such as four wheels, a speed limit, a range of miles, etc. In this case, the class of car is represented by the wheels, the speed limitations, and the mileage.

Inheritance

When one object acquires all the properties and behaviours of parent object i.e. known as inheritance. It provides code reusability. It is used to achieve runtime polymorphism.

1. Sub class - Subclass or Derived Class refers to a class that receives properties from another class.
2. Super class - The term "Base Class" or "Super Class" refers to the class from which a subclass inherits its properties.
3. Reusability - As a result, when we wish to create a new class, but an existing class already contains some of the code we need, we can generate our new class from the old class thanks to inheritance. This allows us to utilize the fields and methods of the pre-existing class.

Polymorphism

When one task is performed by different ways i.e. known as polymorphism. For example: to convince the customer differently, to draw something e.g. shape or rectangle etc. Different situations may cause an operation to behave differently. The type of data utilized in the operation determines the behavior.

Abstraction

Hiding internal details and showing functionality is known as abstraction. Data abstraction is the process of exposing to the outside world only the information that is absolutely necessary while concealing implementation or background information. For example: phone call, we don't know the internal processing.

**(4.1.3) What is the difference between OOP and POP?**

## Differences between OOP and POP:

|  |  |  |
| --- | --- | --- |
| Parameter | Object Oriented Programming(OOP) | Procedure Oriented Programming(POP) |
| Definition | OOP, refers to Object Oriented Programming and its deals with objects and their properties. | POP, refers to Procedural Oriented Programming and its deals with programs and functions. |
| Approach | An object-oriented program uses the Bottom-up approach. | A procedure-oriented program uses the Top-down approach. |
| Access Control | Access control is supported by the means of access modifiers. The access specifiers such as public, private and protected are used. | No access modifiers are supported. |
| Data Hiding | Data can be hidden using Encapsulation | There is no data hiding mechanism. Data is globally accessible. |
| Entity Linkage | Object functions are linked through message passing. | Parameter passing is involved for message passing. |
| Polymorphism | Method Overloading and Method overriding are used in OOP to achieve polymorphism. | POP does not support polymorphism. |
| Virtual Function and Inheritance | OOP supports inheritance and virtual functions and virtual classes via it. | There is no concept of inheritance in POP and neither does it allow it to support the use of virtual classes or virtual functions |
| Code Reusability | OOP supports code reusability. | No Code Reusability is provided by POP. |
| Operator Overloading | It is allowed in OOP | Operator overloading is not allowed |