**Q.2 What is OOP? List OOP concepts?**

**What is OOP** :

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.

**List OOP Concepts :**

Class

Object

Inheritance

Encapsulation

Abstraction

Polymorphism

Class :

Class is a collection of a data member and a member function. A class is a user-defined data type that we can use in our program, and it works as an object constructor, or a "blueprint" for creating objects.

Object :

In C++, Object is a real world entity, for example, chair, car, pen, mobile, laptop etc. In other words, object is an entity that has state and behaviour. Here, state means data and behaviour means functionality. Object is an instance of a class. All the members of the class can be accessed through object.

**Inheritance :**

The capability of a class to derive properties and characteristics from another class is called Inheritance. Inheritance is one of the most important features of Object-Oriented Programming.

Encapsulation :

The meaning of Encapsulation, is to make sure that "sensitive" data is hidden from users. To achieve this, you must declare class variables/attributes as private (cannot be accessed from outside the class). If you want others to read or modify the value of a private member, you can provide public get and set methods.

**Abstraction :**

Abstraction is the process of only showing the necessary details to the user and hiding the other details in the background.

**Polymorphism :**

Polymorphism means "many forms", and it occurs when we have many classes that are related to each other by inheritance.

**Q.3 What is the difference between OOP and POP?**

**OOP :**

OOP treats data as a critical element in the program development and does not allow it to flow freely around the system.

In OOP, the major emphasis is on data rather than procedure (function).

It ties data more closely to the function that operate on it, and protects it from accidental modification from outside function.

OOP allows decomposition of a problem into a number of entities called objects and then builds data and function around these objects.

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

C++, Java, Dot Net, Python etc are the example of Object oriented programming (OOP) language.

POP :

In the procedure oriented approach, large programs are divided into smaller programs known as functions.

In POP, a program is written as a sequence of procedures or function.

In POP, each procedure (function) contains a series of instructions for performing a specific task.

During the program execution each procedure (function) can be called by the other procedures.

To call a procedure (function), we have to write function name only.

While we concentrate onto the development of functions, we give very little attention to the data that are being used by various functions.

In POP, the major emphasis is on procedure (function) and not on the data.

In a multi-function program, many important data items are placed as global so that they may be accessed by all the functions. Each function may have its own local data.

Global data are more vulnerable to an accidental change by a function. In a large program it is very difficult to identify what data is used by which function.