SOFTWARE ENGINEERING ASSIGNMENT
MODULE: 4.1 (C++ Basic)

Q1. WAP to print “Hello World” using C++.
Ans.

```
#include<iostream>
using namespace std;
int main()
{
    cout<<"Hello World";
}
```

Output : Hello World

Q2. What is OOP? List OOP concepts.
Ans.

- Object-Oriented Programming is a methodology or paradigm to design a program using classes and objects.

❖ **List OOP concepts is describe below:**

➢ **Class:** A class is a blueprint or template for creating objects. It defines the attributes (data members) and methods (functions) that the objects of the class will have.

➢ **Object:** An object is an instance of a class. It is a concrete representation of the data and behaviour defined by the class. Objects can be created from

a class, and multiple objects of the same class can coexist.

- **Encapsulation:** Encapsulation is the practice of bundling data (attributes) and methods (functions) that operate on that data into a single unit called a class. It enforces data hiding and access control by specifying access modifiers like public, private, and protected.

- **Inheritance:** Inheritance is a mechanism that allows a class (subclass or derived class) to inherit the properties and behaviours of another class (superclass or base class). It promotes code reusability and hierarchy.

- **Polymorphism:** Polymorphism means "many forms." It allows objects of different classes to be treated as objects of a common base class. Polymorphism enables functions or methods to behave differently based on the actual object type (method overloading and overriding).

- **Abstraction:** Abstraction is the process of simplifying complex reality by modeling classes based on the essential features while hiding the unnecessary details. It focuses on what an object does rather than how it does it.

- **Constructor:** A constructor is a special member function in a class used for initializing the object's state when it's created. C++ supports multiple types of constructors, including default constructors, parameterized constructors, and copy constructors.

- **Destructor:** A destructor is a special member function used to clean up resources and release memory when an object goes out of scope or is explicitly destroyed. It is the counterpart to the constructor.

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

OOP:

- Organizes data into objects, which bundle data (attributes) and behavior (methods) together.
- Focuses on modeling real-world entities as objects and emphasizes the interaction between objects.
- Uses abstraction to hide complex implementation details and expose a simplified interface.
- Promotes code reusability through features like inheritance and polymorphism.
- Encapsulation is a fundamental concept that enforces data hiding and access control.

POP:

- Organizes data into separate data structures and functions that operate on that data.
- Focuses on procedures or functions and the sequence of operations to achieve a task.
- Does not emphasize abstraction to the same extent; it tends to expose implementation details more directly.
- Code reusability may be achieved through functions, but it's typically not as structured as OOP.
- Data is often less encapsulated, and functions may operate directly on data structures.