**ASSIGNMENT: 1**

1. Four Principles of Object Oriented Programming

The four major principles in Object-oriented programming are,

* Encapsulation
* Abstraction
* Inheritance
* Polymorphism

1. Encapsulation: Encapsulation is the process of hiding of data implementation by restricting access to accessors and mutators. An accessor is a method which is used to ask an object about itself. A Mutator is a method that is used to modify the state of an object.
2. Abstraction: Data abstraction is the implementation of an object that contains the same essential properties and actions we can find in the original object we are representing.
3. Inheritance: Inheritance is a way to reuse code of existing objects, depending upon programming language support. In terms of classes, classes can inherit attributes and behaviour from pre-existing classes called base classes and the resulting classes are known as superclass. A base class is a derivative class that inherits one or more properties from another class called superclass. The superclass establishes a common interface and foundational functionality, which specialized subclasses can inherit, modify, and supplement.
4. Polymorphism: Polymorphism clearly describes that multiple methods all with the same name, but slightly different functionality. There two types of polymorphism, Overriding and Overloading. Overriding is also called run-time polymorphism. In this method, the compiler determines which method will be executed, and this decision is made when the code gets compiled. Overloading is also known as compile-time polymorphism. This method will be used for method overriding is determined at runtime based on the dynamic type of an object.(refer: google)
5. Three key benefits of object oriented programming principles

The three key benefits of object oriented programming principles are

* Maintainable
* Reusable
* Scalable

1. Maintainable: Object oriented programming methods make code more maintainable. Easily identify the source of errors because objects are encapsulated. A good OOP design contribute to an application’s maintainability.
2. Reusable: This feature makes it easy to reuse code in new systems. Messages  
   provide a predefined interface to an object’s data and functionality.
3. Scalable: OO applications are more scalable than their structured programming roots. As an object’s interface provides a roadmap for reusing the  
   object in new software, it also provides you with all the information  
   you need to replace the object without affecting other code. This makes  
   it easy to replace old and aging code with faster algorithms and newer  
   technology. (refer: google)