

# COMPUTER SCIENCE

11

## ASSIGNMENT – 5 OBJECT ORIENTED PROGRAMMING WITH C++



**J. KAVITHA**, B.Sc, B.Ed, M.C.A, M.Phil.,  
**Computer Instructor Gr - I**  
GHSS, S.S.KULAM  
Coimbatore – 641107.

# OBJECT ORIENTED PROGRAMMING WITH C++

## Introduction to Object Oriented Programming Techniques:

- Object-Oriented Programming (OOP) is the term used to describe a programming approach based on classes and objects. The object-oriented paradigm allows us to organize software as a collection of objects that consist of both data and behaviour.

## Programming Paradigms:

- Paradigm means organizing principle of a program. It is an approach to programming. There are different approaches available for problem solving using computer. They are Procedural programming, Modular Programming and Object Oriented Programming.

## Procedural programming:

- Procedural means a list of instructions were given to the computer to do something. Procedural programming aims more at procedures. This emphasis on doing things.

## Important features of procedural programming:

- Programs are organized in the form of subroutines or sub programs
- All data items are global. Suitable for small sized software application.
- Difficult to maintain and enhance the program code as any change in data type needs to be propagated to all subroutines that use the same data type. This is time consuming. Example: **FORTRAN** and **COBOL**.

## Modular programming:

- Modular programming consist of a list of instructions that instructs the computer to do something. But this **Paradigm consists of multiple modules, each module has a set of functions of related types. Data is hidden under the modules.** Arrangement of data can be changed only by modifying the module

## Important features of Modular programming

- Emphasis on algorithm rather than data.
- Programs are divided into individual modules.
- Each modules are independent of each other and have their own local data.
- Modules can work with its own data as well as with the data passed to it. Example: **Pascal** and **C**

## Object Oriented Programming:

- Object Oriented Programming paradigm emphasizes on the data rather than the algorithm. It implements programs using **classes** and **objects**.
- **Class:** A Class is a construct in C++ which is used to bind data and its associated function together into a single unit using the encapsulation concept. Class is a user defined data type. Class represents a group of similar objects.
- **Objects:** Represents data and its associated function together into a single unit. Objects are the basic unit of OOP. Basically an object is created from a class. They are instances of class also called as class variables.

## Important features of Object oriented programming:

- Emphasizes on data rather than algorithm.
- Data abstraction is introduced in addition to procedural abstraction.
- Data and its associated operations are grouped in to single unit.
- Programs are designed around the data being operated.
- Relationships can be created between similar, yet distinct data types. Example: **C++**, **Java**, **VB.Net**, **Python** etc.

## Basic Concepts of OOP:

- The Object Oriented Programming has been developed to overcome the drawbacks of procedural and modular programming. It is widely accepted that object-oriented programming is the most important and powerful way of creating software.

## The Object-Oriented Programming approach mainly encourages:

- **Modularisation:** where the program can be decomposed into **modules**.
- **Software re-use:** where a program can be composed from existing and new modules.

## Main Features of Object Oriented Programming:

### Encapsulation :

- The mechanism by which the data and functions are bound together into a single unit is known as **Encapsulation**. It implements abstraction.
- Encapsulation is about binding the data variables and functions together in class. It can also be called **data binding**.
- This encapsulation of data from direct access by the program is called data hiding or information hiding.

### Data Abstraction:

- Abstraction refers to showing only the essential features without revealing background details. Classes use the concept of abstraction to define a list of abstract attributes and function which operate on these attributes.

### Modularity:

- Modularity is designing a system that is divided into a set of functional units (named modules) that can be composed into a larger application.

### Inheritance:

- Inheritance is the technique of building new classes (**derived class**) from an existing Class (**base class**). The most important advantage of inheritance is **code reusability**.

### Polymorphism:

- Polymorphism is the ability of a message or function to be displayed in more than one form.

### Advantages of OOP:

- **Re-usability:** “Write once and use it multiple times” you can achieve this by using class.
- **Redundancy:** Inheritance is the good feature for data redundancy. If you need a same functionality in multiple class you can write a common class for the same functionality and inherit that class to sub class.
- **Easy Maintenance:** It is easy to maintain and modify existing code as new objects can be created with small differences to existing ones.
- **Security:** Using data hiding and abstraction only necessary data will be provided thus maintains the security of data.

### Disadvantages of OOP:

- **Size:** Object Oriented Programs are much larger than other programs.
- **Effort:** Object Oriented Programs require a lot of work to create.
- **Speed:** Object Oriented Programs are slower than other programs, because of their size.



**LEARNING IS A**

**TREASURE**



**THAT WILL FOLLOW**

**ITS OWNER EVERYWHERE.**



**J. Kavitha B.Sc.,B.Ed.,M.C.A.,M.Phil.,**

**Computer Instructor Gr-1**

**GHSS, Sarkarsamakulam**

**Coimbatore - 641107.**