Object Oriented Programming using C++(Unit-1)

- Introduction to OOP
- Procedural Vs. Object Oriented Programming
- Principles of OOP
- Benefits of OOP
- Applications of OOP

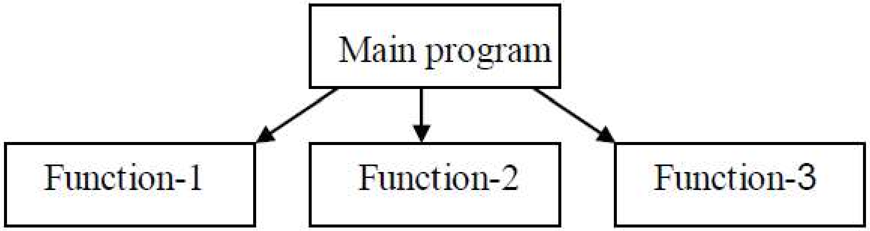
Prepared by: Anil Kumar Tailor, Asst. Prof., Engineering College
Ajmer

Introduction to OOP

- Object-Oriented Programming languages that use objects in programming. Object-oriented programming aims to implement real-world entities like person, a place, a bank account, a table of data etc. in programming. The main aim of OOP is to bind together the data and the functions that operate on them so that no other part of the code can access this data except that function.

Procedural Vs. Object Oriented Programming

- Procedure oriented programming basically consist of writing a list of instruction or actions for the computer to follow and organizing these instruction into groups known as functions.


Main program

Function-1

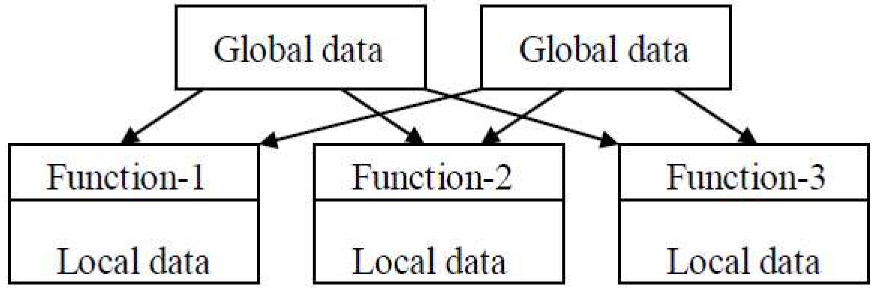
Function-2

Function-3

Procedural Vs. Object Oriented Programming

The disadvantage of the procedure oriented programming languages is:

1. Global data access
2. It does not model real word problem very well
3. No data hiding



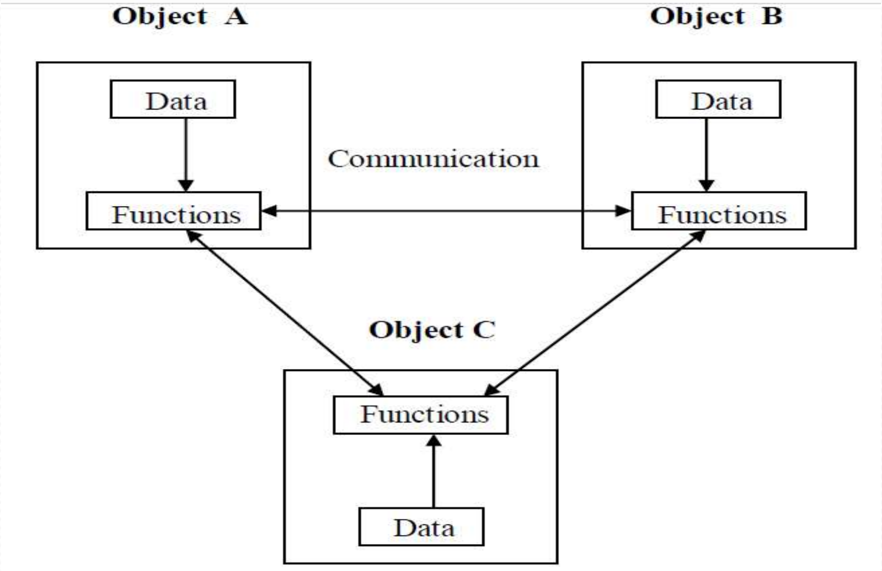
Procedural Vs. Object Oriented Programming

- Characteristics of procedure oriented programming:

1. Emphasis on functions rather than data
2. Large programs are divided into smaller programs known as functions.
3. Most of the functions share global data
4. Data move openly around the system from function to function
5. Function transforms data from one form to another.

Procedural Vs. Object Oriented Programming

- Object-Oriented Programming languages that use objects in programming.



Procedural Vs. Object Oriented Programming

- Characteristics of OOP

1. Emphasis on data rather than procedure.
2. Programs are divided into objects
3. Data structures are designed such that they represent the objects.
4. Functions that operate on the data of an object are tied together in the data structure.
5. Data is hidden and can’t be accessed by external functions.
6. Objects may communicate with each other through functions.
7. New data and functions can be easily added.

Principles of OOP

1. **Abstraction:** The act of representing essential features without including the background details or explanations.
2. **Encapsulation:** The wrapping up of data and function into a single unit (called class) is known as encapsulation.
3. **Inheritance:** The process by which objects of one class acquire the properties of another class.
4. **Polymorphism:** The ability to take more than one form. An operation may show different behaviors in different instances.

Benefits of OOP

1. Through inheritance we can eliminate redundant code and extend the use of existing classes.
2. We can build programs in modules that communicate with one another. This leads to saving of development time and higher productivity.
3. The principle of data hiding helps the programmer to build secure programs.
4. It is possible to have multiple instances of an object to co-exist without any interference.
5. It is easy to partition the work in a project based on objects.
6. Object-oriented systems can be easily upgraded from small to large systems.
7. Message passing techniques for communication between objects makes the interface description with external systems much simpler.
8. Software complexity can be easily managed.

Applications of OOP

1. User interface design such as windows.
2. Real – Time systems.
3. Simulation and modeling
4. Object oriented databases.
5. AI and expert systems.
6. Neural networks and parallel programming.
7. Office automation systems.
8. CIM / CAM / CAD system.

Thank You