OBJECT ORIENTED ANALYSIS DESIGN AND PROGRAMMING - CONCEPTS

UNIT-1

Dr. BALAMURUGAN
ASST. PROFESSOR
DEPARTMENT OF COMPUTER SCIENCE
GOVERNMENT ARTS COLLEGE
COIMBATORE
EMAIL: spbalamurugan@rediffmail.com

A Brief History - OOAD

- The first object–oriented language was Simula (Simulation of real systems) that was developed in 1960 by researchers at the Norwegian Computing Center.
- In 1970, Alan Kay and his research group at Xerox PARK created personal computer named Dynabook and the first pure object-oriented programming language (OOPL) - Smalltalk, for programming the Dynabook.
- In the 1980s, Grady Booch published a paper titled OOD sign that mainly presented a design for the programming language, Ada. In the ensuing editions, he extended his ideas to a complete object–oriented design method.
- In the 1990s, Coad incorporated behavioral ideas to object-oriented methods.
- The other significant innovations were Object Modelling Techniques (OMT) by James Rumbaugh and Object-Oriented Software Engineering (OOSE) by Ivar Jacobson.

OBJECT ORIENTED ANALYSIS(OOA)

- Object–Oriented Analysis (OOA) is the procedure of identifying software engineering requirements and developing software specifications in terms of a software system's object model, which comprises of interacting objects.

- Grady Booch has defined OOA as, *"Object-oriented analysis is a method of analysis that examines requirements from the perspective of the classes and objects found in the vocabulary of the problem domain"*.

- The primary tasks in object-oriented analysis (OOA) are −

    1. Identifying objects
    2. Organizing the objects by creating object model diagram
    3. Defining the internals of the objects, or object attributes
    4. Defining the behavior of the objects, i.e., object actions
    5. Describing how the objects interact

OBJECT ORIENTED DESIGN(OOD)

- Object–Oriented Design (OOD) involves implementation of the conceptual model produced during object-oriented analysis.
- In OOD, concepts in the analysis model, which are technology – independent, are mapped onto implementing classes, constraints are identified and interfaces are designed, resulting in a model for the solution domain.
- Grady Booch has defined object-oriented design as *"a method of design encompassing the process of OO decomposition and a notation for depicting both logical and physical as well as static and dynamic models of the system under design"*.
- OOD deals with implementations:

    1. Restructuring the class data (if necessary),
    2. Implementation of methods, i.e., internal data structures and algorithms,
    3. Implementation of control, and
    4. Implementation of associations.

OBJECT ORIENTED PROGRAMMING(OOP)

- Object-oriented programming (OOP) is a programming paradigm based upon objects (having both data and methods) that aims to incorporate the advantages of modularity and reusability.
- Object-oriented programming languages are C++, Java, Smalltalk, Delphi, C#, Perl, Python, Ruby, and PHP.
- Grady Booch has defined object–oriented programming as *"a method of implementation in which programs are organized as cooperative collections of objects, each of which represents an instance of some class, and whose classes are all members of a hierarchy of classes united via inheritance relationships"*.
- The important features of object–oriented programming are −

1. Bottom–up approach in program design
2. Programs organized around objects, grouped in classes
3. Focus on data with methods to operate upon object's data
4. Interaction between objects through functions
5. Reusability of design through creation of new classes by adding features to existing classes

OBJECT ORIENTED PROGRAMMING(OOP)

- **C++ programming language** was developed in 1980 by Bjarne Stroustrup at bell laboratories of AT&T (American Telephone & Telegraph), located in U.S.A.
- **Bjarne Stroustrup** is known as the **founder of C++ language.**
- It was develop for adding a feature of **OOP (Object Oriented Programming)** in C without significantly changing the C component.
- C++ programming is "relative" (called a superset) of C, it means any valid C program is also a valid C++ program.
- Basic concepts and terminologies of object–oriented systems.

1. Objects and Classes
2. Encapsulation & Data hiding
3. Message passing
4. Inheritance
5. Polymorphism
6. Generalization and Specialization
7. Links and Association
8. Aggregation and Composition

RELATIONSHIP AMONG OBJECTS/CLASSES

- A **link** represents a connection through which an object collaborates with other objects.
- Rumbaugh has defined it as “a physical or conceptual connection between objects”.
- Through a link, one object may invoke the methods or navigate through another object.
- A link depicts the relationship between two or more objects.
- **Association** is a group of links having common structure and common behavior.
- Association depicts the relationship between objects of one or more classes.
- A link can be defined as an instance of an association.
- **Degree of an association** denotes the number of classes involved in a connection. Degree may be unary, binary, or ternary.
  1. **unary relationship** connects objects of the same class.
  2. **binary relationship** connects objects of two classes.
  3. **ternary relationship** connects objects of three or more classes.

RELATIONSHIP AMONG OBJECTS/CLASSES

- Cardinality of a binary association denotes the number of instances participating in an association. There are three types of cardinality ratios, namely −
    1. **One–to–One** − A single object of class A is associated with a single object of class B.
    2. **One–to–Many** − A single object of class A is associated with many objects of class B.
    3. **Many–to–Many** − An object of class A may be associated with many objects of class B and conversely an object of class B may be associated with many objects of class A.

RELATIONSHIP AMONG OBJECTS/CLASSES

- Aggregation or composition is a relationship among classes by which a class can be made up of any combination of objects of other classes.
- It allows objects to be placed directly within the body of other classes.
- Aggregation is referred as a “part–of” or “has–a” relationship, with the ability to navigate from the whole to its parts.
- An aggregate object is an object that is composed of one or more other objects.
- **Example -** In the relationship, “a car has–a motor”, car is the whole object or the aggregate, and the motor is a “part–of” the car.
  Aggregation may denote −

  1. **Physical containment** − Example, a computer is composed of monitor, CPU, mouse, keyboard, and so on.
  2. **Conceptual containment** − Example, shareholder has–a share.

BENEFITS OF OBJECT MODEL

- It helps in faster development of software.
- It is easy to maintain.
- Suppose a module develops an error, then a programmer can fix that particular module, while the other parts of the software are still up and running.
- It supports relatively hassle-free upgrades.
- It enables reuse of objects, designs, and functions.
- It reduces development risks, particularly in integration of complex systems.

References

- https://www.tutorialspoint.com/object_oriented_analysis_design/ooad_object_oriented_model.htm
- https://www.geeksforgeeks.org/object-oriented-programming-in-cpp
- https://www.javatpoint.com/cpp-oops-concepts

TEXT BOOKS

1) Grady Booch, “Object Oriented Analysis and Design with Applications”, Second Edition, Pearson Education.
2) Ashok N. Kamthane, “Object Oriented Programming with ANSI &Turbo C++” , First Indian Print, Pearson Education, 2003.

REFERENCE BOOKS

1) Balagurusamy “Object Oriented Programming with C++”, TMCH, Second Edition, 2003.