

Object oriented programming

Second Grade

Assist. Lect. Ali Haitham

Introduction

Programming is a creative process carried out by programmers to instruct a computer on how to do a task.

A **program** is a set of instructions that tells a computer what to do in order to come up with solution to a particular problem.

The approaches to the programming process referred to as **programming paradigms**

Programming Paradigms

- Programming paradigm is a way to classify programming languages according to their style of programming and features they provide.
- There are several features that determine a programming paradigm such as modularity, objects, interrupts or events, control flow etc. A programming language can be single paradigm or multi-paradigm

Unstructured Programming

- Unstructured programming is the oldest paradigm and is still in practice.
- It mainly focuses on steps to be done and works on the logic of “First do this then do that”. operations must take place.
- It defines a sequence of statements in order of which the
 - In unstructured programming, the control flow is explicit and depend on collection of GOTO statements.
- unstructured programming lacks the support of modularity.

Structured Programming

- It can be defined as a programming approach in which the program is made as a single structure. It means that the code will execute the instruction by instruction one after the other.
- It doesn't support the possibility of jumping from one instruction to some other with the help of any statement like GOTO, etc.
- The instructions in this approach will be executed in a serial and structured manner. The languages that support Structured programming approach are: C , C++ , Java and C#

Example structured programming vs unstructured programming

program fragments, the first is structured, while the second

Structured:

```
IF x<=y THEN
  BEGIN
    z := y-x;
    q :=SQRT(z);
  END
ELSE
  BEGIN
    z := x-y;
    q := -SQRT(z)
  END;
WRITELN(z,q);
```

Unstructured:

```
IF x>y THEN GOTO 2;
z := y-x;
q := SQRT(z);
GOTO 1;
2: z:= x-y;
q:=-SQRT(z);
1: writeln(z,q);
```

Procedure Oriented Programming

- ❑ Conventional programming language using high level language such as Cobol and C, is commonly known as Procedure Oriented Programming (POP).
- ❑ In Procedure Oriented Programming, the problem is viewed as a sequence of a thing to be done.
 - ❑ The primary focus is on functions.
 - ❑ Procedure Oriented Programming basically consists of writing a list of instructions for the computer to follow, and organizing these functions into groups known as functions.
 - ❑ In a multi-function program, many important data items are placed as global so that they may be accessed by all the functions.
 - ❑ Each function may have its own local data.

Object Oriented Programming

Object Oriented Programming is the most recent concept among programming model. The motivating factor in the invention of object oriented approach is to remove some of the flows encountered in the procedural approach. OOPS treats data as a critical element in the program development and does not allow it to flow freely around the system. It binds data more closely to the functions that operate on it, and protects it from accidental modification from outside functions. Some of the striking features of Object Oriented Programming are:

- Importance on data rather than procedure.
- Programs are divided into what are known as objects.
- Data structures are designed as such that they characterize the objects.

Benefits of OOP

- 1) As OOP is closer to the real world phenomena, hence, it is easier to map real world problems onto a solution in OOP.
- 2) The objects in OOP have the state and behavior that is similar to the real world objects.
- 3) It is more suitable for large projects.
- 4) The projects executed using OOP techniques are more reliable.
- 5) It provides the bases for increased testability (automated testing) and hence higher quality.
- 6) **Abstraction** techniques are used to hide the unnecessary details and focus is only on the relevant part of the problem and solution.
- 7) **Encapsulation** helps in concentrating the structure as well as the behavior of various objects in OOP in a single enclosure.
- 8) Object oriented systems are easier to upgrade/modify.

Some of the most popular Object-oriented Programming languages are :

- a) C++**
- b) Ruby**
- c) Java.**
- d) Delphi**
- e) smalltalk**
- f) Charm++**
- g) Eiffle.**
- h) Simula.**

Applications of OOPS

Object concept helps to translate our thoughts to a program. It provides a way of solving a problem in the same way as a human being perceives a real world problem and finds out the solution. It is possible to construct large reusable components using object-oriented techniques. Development of reusable components is rapidly growing in commercial software industries. If there is complexity in software development, object-oriented programming is the best paradigm to solve the problem. The following areas make the use of OOP:

1. Image Processing
2. Pattern Recognition
3. Computer Assisted Concurrent Engineering
4. Computer Aided Design and Manufacturing
5. Computer Aided Teaching
6. Intelligent Systems

- 
7. Database Management Systems
 8. Web-based Applications
 9. Distributed Computing and Applications
 10. Component-based Applications
 11. Business Process Re-engineering
 12. Enterprise Resource Planning
 13. Data security and management
 14. Mobile Computing
 15. Data Warehouse and Data Mining
 16. Parallel Computing

Comparison of Procedure Oriented Programming and Object Oriented Approach

	Procedure Oriented Programming	Object Oriented Programming
Divided Into	Program is divided into small parts called functions .	Program is divided into parts called objects .
Importance	Importance is not given to data but to functions	Importance is given to the data rather than functions(methods)
Approach	Follows Top Down approach .	Follows Bottom Up approach .
Access Specifiers	does not have any access specifier.	has access specifiers named Public, Private, Protected, etc.
Data Moving	data can move freely from function to function	objects can move and communicate with each other through member functions.

Expansion	To add new data and function in POP is not so easy.	provides an easy way to add new data and function.
Data Access	most function uses global data for sharing that can be accessed freely from function to function.	data can not move easily from function to function, it can be kept public or private so we can control the access of data.
Data Hiding	does not have any proper way for hiding data so it is less secure .	provides data hiding so provides more security .
Overloading	overloading is not possible.	overloading is possible
Examples	Example of POP are : C, VB, FORTRAN, Pascal.	Example of OOP are : C++, JAVA, VB.NET, C#.NET.

Fundamental Principles of OOP

In order for a programming language to be object-oriented, it has to enable working with classes and objects as well as the implementation and use of the fundamental object-oriented principles and concepts: inheritance, abstraction, encapsulation and polymorphism:

- **Encapsulation** Is a process of wrapping code and data together into a single unit
- **Inheritance** is the mechanism of making new classes from existing one.
- **Abstraction** is a process of hiding the implementation details and showing only functionality to the user.
- **Polymorphism** is a concept by which we can perform a single action by different ways.

Classes and Objects

The two most important concepts in object-oriented programming are the class and the object.

An object is comprised of data and operations that manipulate these data. For example, a Student object may consist of data such as name, gender, birth date, home address, phone number, and age and operations for assigning and changing these data values.

For the computer to be able to create an object, we must provide a definition, called a class. A class is a kind of template that dictates what objects can and cannot do. An object is called an instance of a class. Once a class is defined, we can create as many instances of the class as a program requires. A class must be defined before you can create an instance (object) of the class.

differences between object and class

Object	Class
Object is an instance of a class.	Class is a blueprint or template from which objects are created.
Object is a real world entity such as pen, laptop, mobile, bed, keyboard, mouse, chair etc.	Class is a group of similar objects.
Object is a physical entity.	Class is a logical entity.
Object is created through new keyword mainly e.g. Student s1=new Student();	Class is declared using class keyword e.g. class Student{}
Object is created many times as per requirement.	Class is declared once.
Object allocates memory when it is created.	Class doesn't allocated memory when it is created.
There are many ways to create object in java such as new keyword, newInstance() method, clone() method, factory method and deserialization.	There is only one way to define class in java using class keyword.

method and function

A **function** is a combination of instructions that are combined to achieve some results, it requires some input (called arguments) and returns some results.

A **method** in object-oriented programming is a procedure associated with a class, it defines the behavior of the objects(or an action that an object is able to perform). A method defined for a class is called a class method.

How does function differ from a method?

Ans.: A function is independent and not associated with a class. You can use this function anywhere in your code, and you don't need to have an object to use it.

Advantages of OOP

1. Code reusability in terms of inheritance.
2. Object-oriented system can be easily upgraded from one platform to another.
3. Complex projects can be easily divided into small code functions.
4. The principle of abstraction and encapsulation enables a programmer to build secure programs.
5. Software complexity decreases.
6. Principle of data hiding helps programmer to design and develop safe programs.
7. Rapid development of software can be done in short span of time.
8. More than one instance of same class can exist together without any interference.