Mo Tu We Th Fr Sa Su Date / /

Evolution of OOPs :-

→ • Initially for writing small and simple programs, Machine language was introduce
• Next came the Assembly Language which was used for designing larger programs.
• Both machine and Assembly languages are machine dependent.
• Next came Procedural Programming Approach which ~~enabled~~ helps us to write larger and hundred lines of codes.
• Then in 1970, structured Programming Approach was developed for ~~does~~ writing medium sized programs.
• In 1980, size of programs increase rapidly, Therefore, new approach known as OOP was invented.

Mo Tu We Th Fr Sa Su Date / /

OOPs

- OOPs was introduced to overcome the drawbacks of procedural programming language such as lack of reusability and maintainability.

- Basicly, OOPs is a way of solving complex problems by breaking them into smaller problems using objects.

- In OOPs we write programs using classes and objects with the help of OOPs features such as abstraction, encapsulation, inheritance and polymorphism.

- Data of an object can only be accessed by the function which is associated with it.

Mo Tu We Th Fr Sa Su Date / /

- Communication between objects is done through objects.

Advantages of OOPs :-

- Emphasis on data rather than procedure
- Programs are divided into entities known as objects.
- In data structure, Function that operate on data of an objects are tied together.
- Data is hidden and cannot be accessed by external functions.

Mo Tu We Th Fr Sa Su Date / /

1) OOPs Concepts

↳ Class
↳ Objects
↳ Encapsulation
↳ Abstraction
↳ Polymorphism
↳ Inheritance

* Objects :-

- An Object is an instance of a class. When a class is defined, no memory is allocated but when it is initiated, memory is allocated.
- Objects are the basic run time entities of any object oriented system.

Mo Tu We Th Fr Sa Su Date / /

* Class :-

- It is a user-defined data type, which holds its own data members and member functions, which can be accessed and used by creating an object of that class

* Encapsulation :-

- It is the process of binding data members (variables, properties) and member functions (methods) into a single unit
- This is to avoid the access of private data members from outside the class.

Mo Tu We Th Fr Sa Su Date / /

* Abstraction :-

- Data abstraction is one of the most essential and important features of object oriented programming in C++

- Abstraction means displaying only essential information and hiding the details or implementation.

* Polymorphism :-

meaning ↓

- It is a greek word → having more than one form

- It can be define as the ability of a message to be displayed in more than one form.

Mo Tu We Th Fr Sa Su Date / /

* Inheritance :-

- The capability of a class to derive properties and characteristics from another class is called Inheritance.

Sub Class :- The class that inherits properties from another class is called Sub class or Derived Class.

Super Class :- The class whose properties are inherited by sub class is called Base Class or Super class.

Mo Tu We Th Fr Sa Su Date / /

Functional Programming

1) The FPL supports immutable data (values and data can't be changed)
2) FPL supports the Declarative Programming Model.
3) FPL focuses on the "What we are doing"
4) The methods of FPL will not produce any side-effects
5) FPL follows parallel programming
6) for the flow control, we do function calls & function calls with recursion
7) for iteration of data collection FPL use "Recursion" concept.
8) In FPL, execution of the statements in the order is not important.
9) FPL supports "Abstraction over Data" and "Abstraction over Behavior".

Mo Tu We Th Fr Sa Su Date / /

Object-Oriented Programming

1) OOP uses mutable data. (values and Data can be changed)
2) OOP supports the imperative Programming Model
3) OOP focuses on the "How we are doing"
4) Methods of OOP can produce the side-effects.
5) OOP does not work on paralled prog.
6) for the flow control, OOP supports use of the loops and conditional statement.
7) OOP uses the "Loop" concept for the iteration of Data collection.
8) In OOP, execution of the statements in the order is very important.
9) OOP supports Only "Abstraction over Data".

Mo Tu We Th Fr Sa Su Date / /

Why do we need OOP?

OOP was discovered because limitations were found in earlier procedural programming.

Drawbacks :-

1) Every function has complete access to global data.
2) The new programmer can easily corrupt the data.
3) ~~We can access the data of one function from other since there~~
3) lack of protection, we can access the data of one function from other.
4) In large program it is very difficult to identify what data is being used by the function.
5) all functions need to be modified when new data is added.
6) Does not solve real world problem very well.