




Introduction to object-oriented programming

2023-2024



you have seen so far the first programming paradigm based on the expression:

First do this and next do that.

In this module we will see a new and important paradigm:

object-oriented programming. OOP





01

Evolution of Programming?




© **The program:** instructions executed sequentially

1. **Procedural programming:** A procedure (routine, subroutine) or function simply contains a series of steps to carry out. Any procedure can be called at any stage of program execution including itself. (functions: cos, sin ...)
2. **Modular programming:** the program is seen as a set of modules. Modularity consists of dividing the program or application into units without side effects between them: only depends on the parameters explicitly passed as arguments. (Math.h, stdio.h are modules)

The background features a light blue dot grid pattern. Overlaid on this are various abstract circuit-like elements: thin lines in purple, blue, and orange that resemble traces on a PCB, and several small purple circles with white centers. The overall aesthetic is clean and technical.


02

**What is object-
oriented programming?**



Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or objects, rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior.

OOP focuses on the objects that developers want to manipulate rather than the logic required to manipulate them. This approach to programming is well-suited for programs that are large, complex and actively updated or maintained. This includes programs for manufacturing and design, as well as mobile applications; for example, OOP can be used for manufacturing system simulation software.





02

What OOP fundamentals?






OOP Fundamentals

- Object, class and instance**
- Class and instance members**
- encapsulation**
- Inheritance**
- Polymorphism**
- Abstract/concrete class**
- Interfaces**



02

What is OOP advantages?



➤ **Modularity:** objects form compact modules grouping together data and a set of operations.

➤ **Abstraction:** The object entities of the OOP are close to those of the real world.

➤ **productivity and reusability:** The more complex the application, the more interesting the OOP approach is in terms of productivity. The level of reusability is higher than imperative programming.

➤ **safety:** Encapsulation and class typing provide a certain robustness to applications.

