* 10 Most popular programming paradigms: -

1. Functional

Functional programming is a programming paradigm in which we try to bind everything in pure mathematical functions style. It is a declarative type of programming style.

Languages which uses this paradigm are: - Racket, Scala etc.

1. Imperative

It is one of the oldest programming paradigms. It features close relation to machine architecture. It is based on Von Neumann architecture. It works by changing the program state through assignment statements. It performs step by step task by changing state. The focus is on how to achieve the goal. The paradigm consists of several statements and after execution of all the result is stored.

Languages which uses this paradigm are: - C, Fortran, Basic.

1. Object Oriented

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. Objects, which are usually instances of classes, are used to interact with one another to design applications and computer programs.

Languages which uses this paradigm are: - C++, Java etc.

1. Structured

**Structured programming** is a programming paradigm aimed at improving the clarity, quality, and development time of a computer program by making extensive use of the structured control flow constructs of selection (if/then/else) and repetition (while and for), block structures, and subroutines in contrast to using simple tests and jumps such as the go to statement, which can lead to “**spaghetti code**” that is potentially difficult to follow and maintain.

Languages which uses this paradigm are: - Ruby, Perl, C# etc.

1. Procedural

Procedural Programming may be the first programming paradigm that a new developer will learn. Fundamentally, the procedural code is the one that directly instructs a device on how to finish a task in logical steps. This paradigm uses a linear top-down approach and treats data and procedures as two different entities. Based on the concept of a procedure call, Procedural Programming divides the program into procedures, which are also known as routines or functions, simply containing a series of steps to be carried out.

Languages which uses this paradigm are: - ColdFusion, Pascal etc.

1. Event-Driven

Event-driven programming is currently the default paradigm in software engineering. As the name suggests, it uses events as the basis for developing the software. These events can be something the users are doing clicking on a specific button, picking an option from drop-down, typing text into a field, giving voice commands, or uploading a video or system-generated events such as a program loading.

Languages which uses this paradigm are: - C++, JAVA etc.

1. Flow Driven

The flow-based paradigm tries to have a natural way of abstracting logic and an obvious way of visualizing each of its elements. Unlike imperative programming, FBP does not operate on a global memory. In that regard it is very similar to functional programming. But unlike functional programming, the order of its elements is important and has a meaning.

Languages which uses this paradigm are: - Java, Python etc.

1. Logic

Logic programming is a computer programming paradigm in which [program](https://www.computerhope.com/jargon/p/program.htm) statements express facts and rules about problems within a system of formal logic. Rules are written as logical clauses with a head and a body; for instance, "H is true if B1, B2, and B3 are true." Facts are expressed like rules, but without a body; for instance, "H is true."

Languages which uses this paradigm are: - Absys, CHIP etc.

1. Aspect Oriented

Aspect-Oriented Programming (AOP) is a programming paradigm which complements Object-Oriented Programming (OOP) by separating concerns of a software application to improve modularization. The separation of concerns (SoC) aims for making a software easier to maintain by grouping features and behavior into manageable parts which all have a specific purpose and business to take care of.

Languages which uses this paradigm are: - AspectJ, PROSE etc.

1. Constraint

Constraint programming is a powerful paradigm for solving combinatorial search problems that draws on a wide range of techniques from artificial intelligence, operations research, algorithms, graph theory and elsewhere. The basic idea in constraint programming is that the user states the constraints and a general-purpose constraint solver is used to solve them.

Languages which uses this paradigm are: - Prolog,

References

<https://hackr.io/blog/procedural-programming>

<https://searchsoftwarequality.techtarget.com/definition/structured-programming-modular-programming>

<https://www.tutorialspoint.com/object_oriented_analysis_design/ooad_object_oriented_paradigm.htm>

<https://www.geeksforgeeks.org/introduction-of-programming-paradigms/>

<https://www.geeksforgeeks.org/functional-programming-paradigm/>

<http://www.technologyuk.net/computing/software-development/software-design/event-driven-programming.shtml>

<https://www.cse.unsw.edu.au/~tw/brwhkr08.pdf>

<https://medium.com/bitspark/what-the-hell-is-flow-based-programming-d9e88a6a7265>

<https://www.computerhope.com/jargon/l/logic-programming.htm>

<https://flowframework.readthedocs.io/en/stable/TheDefinitiveGuide/PartIII/AspectOrientedProgramming.html>