Procedural Languages

- These are command driven and statement oriented language.
- Program executes in sequence of statements. Syntax :

```
stmt 1;
stmt 2;
stmt 3;
```

- In it, it is very difficult to define user defined data type. It only supports built in data type.

- Procedural Language**:** In procedural languages, the program code is written as a sequence of instructions. User has to specify “what to do” and also “how to do” (step by step procedure). These instructions are executed in the sequential order. These instructions are written to solve specific problems.
- **Examples of Procedural languages:** FORTRAN, COBOL, ALGOL, BASIC, C and Pascal.
- Non-Procedural Language**:**
  In the non-procedural languages, the user has to specify only “what to do” and not “how to do”. It is also known as an applicative or functional language. It involves the development of the functions from other functions to construct more complex functions.
- **Examples of Non-Procedural languages:** SQL, PROLOG, LISP.

Difference between Procedural and Non-Procedural language:

It is command-driven language	It is a function-driven language
It works through the state of machine	It works through the mathematical functions
Its semantics are quite tough.	Its semantics are very simple.
It returns only restricted data types and allowed values	can return any datatype or value
Efficiency is high	Efficiency is less
Size of the program written in Procedural language is large.	Size of the Non-Procedural language programs are small.
It is not suitable for time critical applications.	It is suitable for time critical applications
Iterative loops and Recursive calls both are used in the Procedural languages.	Recursive calls are used in Non-Procedural languages.

Structured programming

- A structured programming language is the language that supports three patterns: sequence (an ordered list of statements), selection (e.g. if statements) and repetition (e.g. loops). The combination of these patterns allows the non-inclusion of unstructured statements that include “goto”.

Logical Programming

- **Logical Programming** is a type of programming paradigm that uses logic circuits to control how facts and rules about the problems within the system are represented or expressed. In it, logic is used to represent knowledge, and inference is used to manipulate it. It tells the model about how to accomplish a goal rather than what goal to accomplish.

Functional Programming	Logical Programming
It is totally based on functions.	It is totally based on formal logic.
In this programming paradigm, program statements usually express or represent facts and rules related to problems within a system of formal logic	In this programming paradigm, programs are constructed by applying and composing functions.
These are specially designed to manage and handle symbolic computation and list processing applications.	These are specially designed for fault diagnosis, natural language processing, planning, and machine learning.
Its main aim is to reduce side effects that are accomplished by isolating them from the rest of the software code.	Its main aim is to allow machines to reason because it is very useful for representing knowledge.
Some languages used in functional programming include Clojure, Wolfram Language, Erland, OCaml, etc.	Some languages used for logic programming include Absys, Cycl, Alice, ALF (Algebraic logic functional programming language), etc.
It reduces code redundancy, improves modularity, solves complex problems, increases maintainability, etc.	It is data-driven, array-oriented, used to expres knowledge, etc.
It usually supports the functional programming paradigm.	It usually supports the logic programming paradigm.
Testing is much easier as compared to logical programming.	Testing is comparatively more difficult as compared to functional programming.

Object Oriented Programming Languages

- Object is defined as a group of procedures that shares a state.
- A collection of objects is grouped into class.
- It has same sequence control statements Procedural Languages except an added exception handling.
- It supports overloading of functions.
- More complex than Procedural Languages.

Comparison of C & C++

C	C++
❖ Procedural Language.	❖ Object Oriented Language.
❖ Simple language.	❖ Not simple language.
❖ Operator overloading is not possible.	❖ Operator overloading is possible.
❖ Function prototyping before body of main are optional.	❖ Function prototyping before body of main is compulsory.
❖ Concept of classes is there.	❖ No concept of classes.

C

- Multiple Inheritane is not used.
- Destructors are not used.
- No use of Scope Resolution.
- Doesn’t uses goto statement.
- No namespaces are present.
- Multiple declaration of global variables are allowed.
- malloc () & calloc() are used for memory allocation.

C++

- Multiple Inheritane is used.
- Destructors are used.
- Use of Scope Resolution.
- Uses goto statement.
- Namespaces are present in it.
- Multiple declaration of global variables are not allowed.
- New & Delete are used for memory allocation.