Programming with C++

Paradigm

Prepared by- Mohd Rashid, Dept. of Computer Science, Career College Bhopal

We are going to learn-

- Definition of Paradigm
- Types of Paradigm
- Imperative Paradigm
- Declarative Paradigm

Learning Objectives

1. Develop understating about different paradigms of programming language.
2. How can we choose programming language as per our problem type.
3. Distinguish programming languages easily.

Paradigm

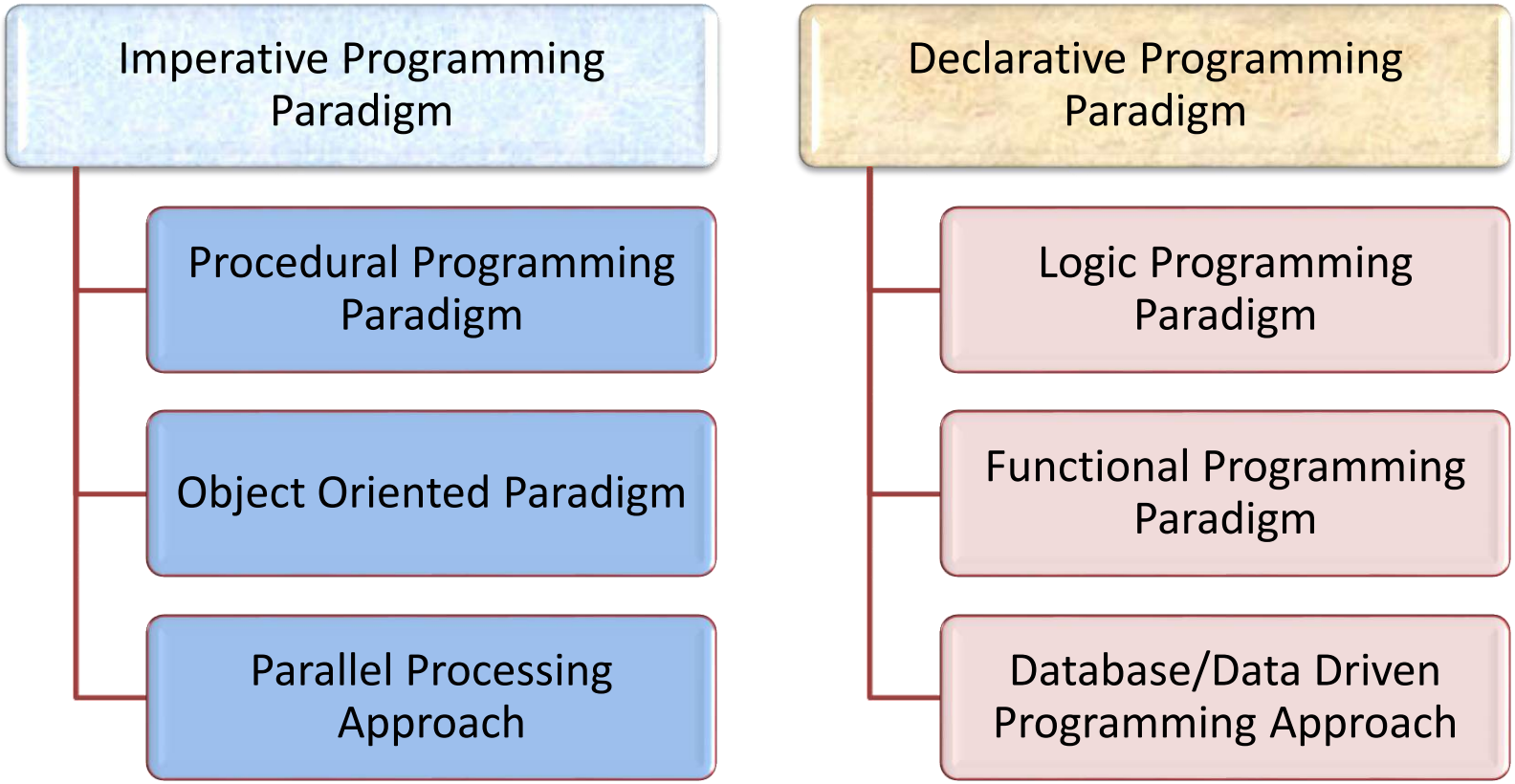
A programming paradigm is a style, or “way,” of programming.

Paradigm can also be termed as method to solve some problem or do some task using tools and techniques that are available to us following some approach.

Programming paradigm is an approach to solve problem using some programming language

There are lots for programming language that are known but all of them need to follow some strategy when they are implemented and this strategy is paradigms.

Programming Paradigms



Imperative Programming Paradigm

Prepared by- Mohd Rashid, Dept. of Computer Science, Career College Bhopal

Imperative Programming Paradigm

It is one of the oldest programming paradigm. It features close relation to machine architecture. It performs step by step task by changing state.

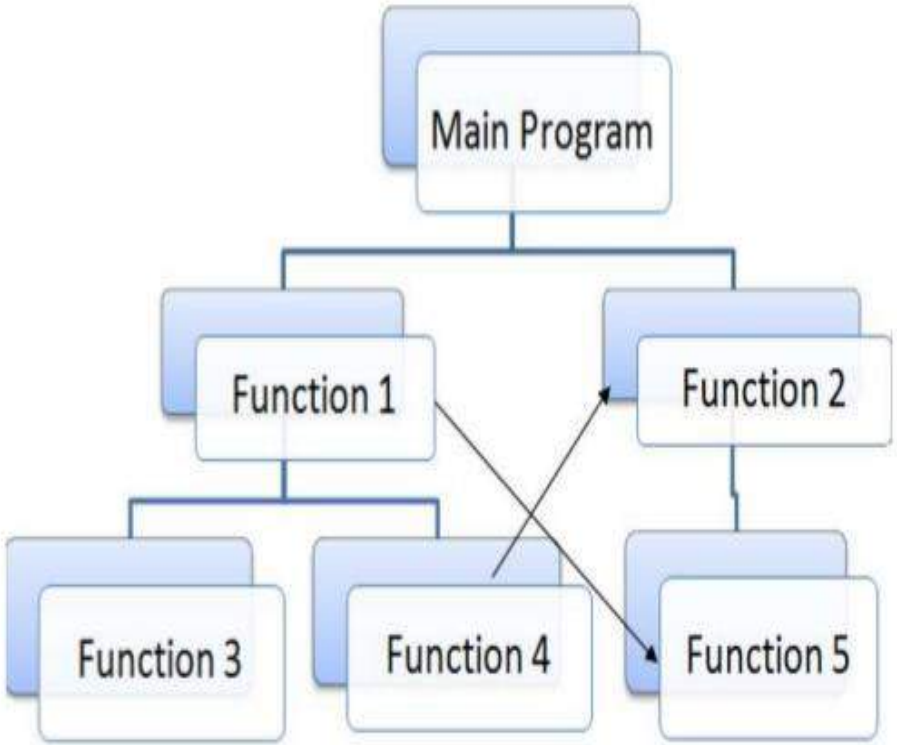
It is based on Von Neumann architecture. It works by changing the program state through assignment statements.

The main focus is on how to achieve the goal. The paradigm consist of several statements and after execution of all the result is stored.

Procedural Programming Paradigm

This paradigm emphasizes on procedure in terms of under lying machine model. It has the ability to reuse the code and it was boon at that time when it was in use because of its reusability.

Examples- C, C++, Java, ColdFusion, Pascal



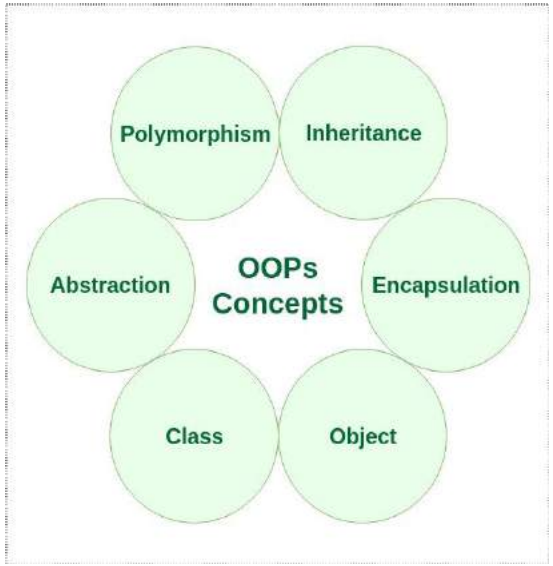
Object Oriented Programming Paradigm

The program is written as a collection of classes and object which are meant for communication. The smallest and basic entity is object and all kind of computation is performed on the objects only.

More emphasis is on data rather procedure. It can handle almost all kind of real life problems which are today in scenario.

Object Oriented concepts are classes, object, inheritance, encapsulation, abstraction and polymorphism.

Examples- C++, C#, Simula, Java, Python, Smalltalk, Objective-C, Visual Basic.Net, Ruby.

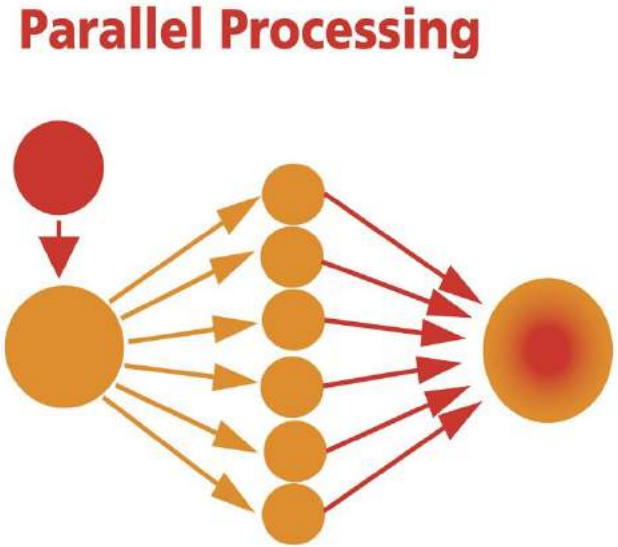


Parallel Processing Approach

Parallel processing is the processing of program instructions by dividing them among multiple processors.

A parallel processing system posses many numbers of processor with the objective of running a program in less time by dividing them. This approach seems to be like divide and conquer.

Examples are NESL (one of the oldest one) and C/C++ also supports because of some library function.



Declarative Programming Paradigm

Prepared by- Mohd Rashid, Dept. of Computer Science, Career College Bhopal

Declarative Programming Paradigm

declarative programming is a style of building programs that expresses logic of computation without talking about its control flow.

It often considers programs as theories of some logic. It may simplify writing parallel programs.

The focus is on what needs to be done rather how it should be done basically emphasize on what code is actually doing.

It just declare the result we want rather how it has be produced. This is the only difference between imperative (how to do) and declarative (what to do) programming paradigms.

Logic Programming Paradigm

It can be termed as abstract model of computation. It would solve logical problems like puzzles, series etc.
In logic programming we have a knowledge base which we know before and along with the question and knowledge base which is given to machine, it produces result.
In normal programming languages, such concept of knowledge base is not available but while using the concept of artificial intelligence, machine learning we have some models like Perception model which is using the same mechanism.
In logical programming the main emphasize is on knowledge base and the problem. The execution of the program is very much like proof of mathematical statement e.g. Prolog

Functional Programming Paradigm

The functional programming paradigms has its roots in mathematics and it is language independent.

The key principal of this paradigms is the execution of series of mathematical functions.

The central model for the abstraction is the function which are meant for some specific computation and not the data structure.

Data are loosely coupled to functions. The function hide their implementation. Function can be replaced with their values without changing the meaning of the program. Some of the languages like perl, javascript mostly uses this paradigm.

Database/Data Driven Programming Approach

This programming methodology is based on data and its movement. Program statements are defined by data rather than hard-coding a series of steps.

A database program is the heart of a business information system and provides file creation, data entry, update, query and reporting functions.

There are several programming languages that are developed mostly for database application. For example SQL. It is applied to streams of structured data, for filtering, transforming, aggregating (such as computing statistics), or calling other programs. So it has its own wide application.