

# Programming Paradigm

WEEK2, LECTURE 3

DR. RIZWAN REHMAN

---

# Programming Paradigm

---

A paradigm is the preferred approach to programming that a language supports, or we can say that its programming style.

It defines the style, capabilities and limitations of a programming approach.

---

Main paradigms we are going to discuss are:

1. Procedural
2. Structured.
3. Object-Oriented.

# Procedural Approach

---

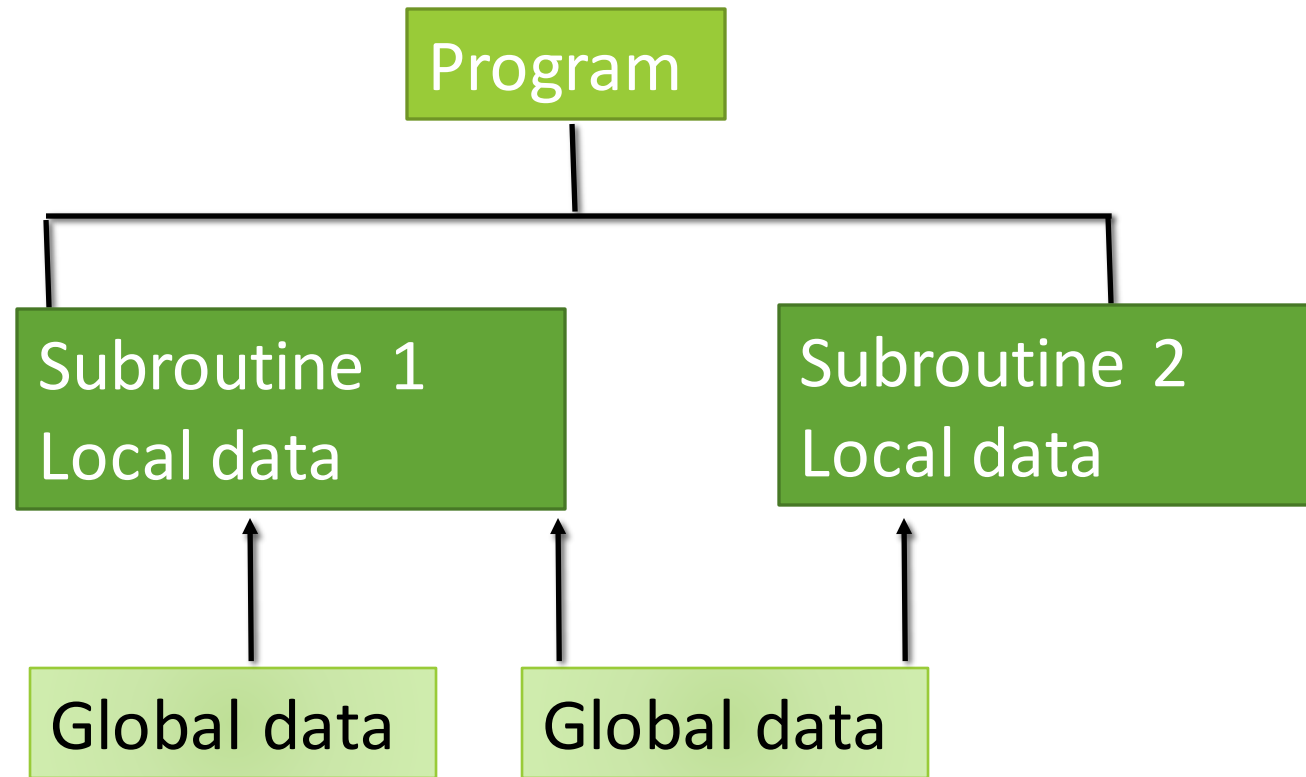
Divide the code in procedures: routines, subroutines, modules methods, or functions.

Procedural programming languages provide instructions to the computer by following a set of steps.

In this approach the program code is divided into group of smaller programs which are also referred to as procedures or subroutines or functions.

The subroutines can operate upon and access their own local data as well as global data shared by all the subroutines

---



---

The developer will begin with the problem (process) when a problem needs to be fixed using procedural programming, and he will then logically break the problem down into subproblems (Sub-Procedures). This method will then be carried out repeatedly up until a sub-procedure is easy enough to complete on its own.

# Advantages

---

1. Division of work.
2. Debugging and testing.
3. Maintainability of code.
4. Reusability.

# Structured Paradigm

---

According to structured programming, an application programme should be broken up into a hierarchy of modules or autonomous elements, some of which may also contain other autonomous elements.

Code may be further organised within each element using blocks of connected logic that are intended to increase readability and maintainability.



---

Structured programming allows the programmer to understand the program easily.

If a program consists of thousands of instructions and an error occurs then it is complicated to find that error in the whole program, but in structured programming, we can easily detect the error and then go to that location and correct it.

Structured programming employs methods using following:

---

1. Top-down analysis for problem solving.
2. Modularization for program structure and organization.
3. Structured code for the individual modules.

# Advantages

---

1. Easier to read and understand.
2. Easier to Maintain.
3. Development is easier as it requires less effort and time.
4. Easier to Debug.
5. Modular Approach.

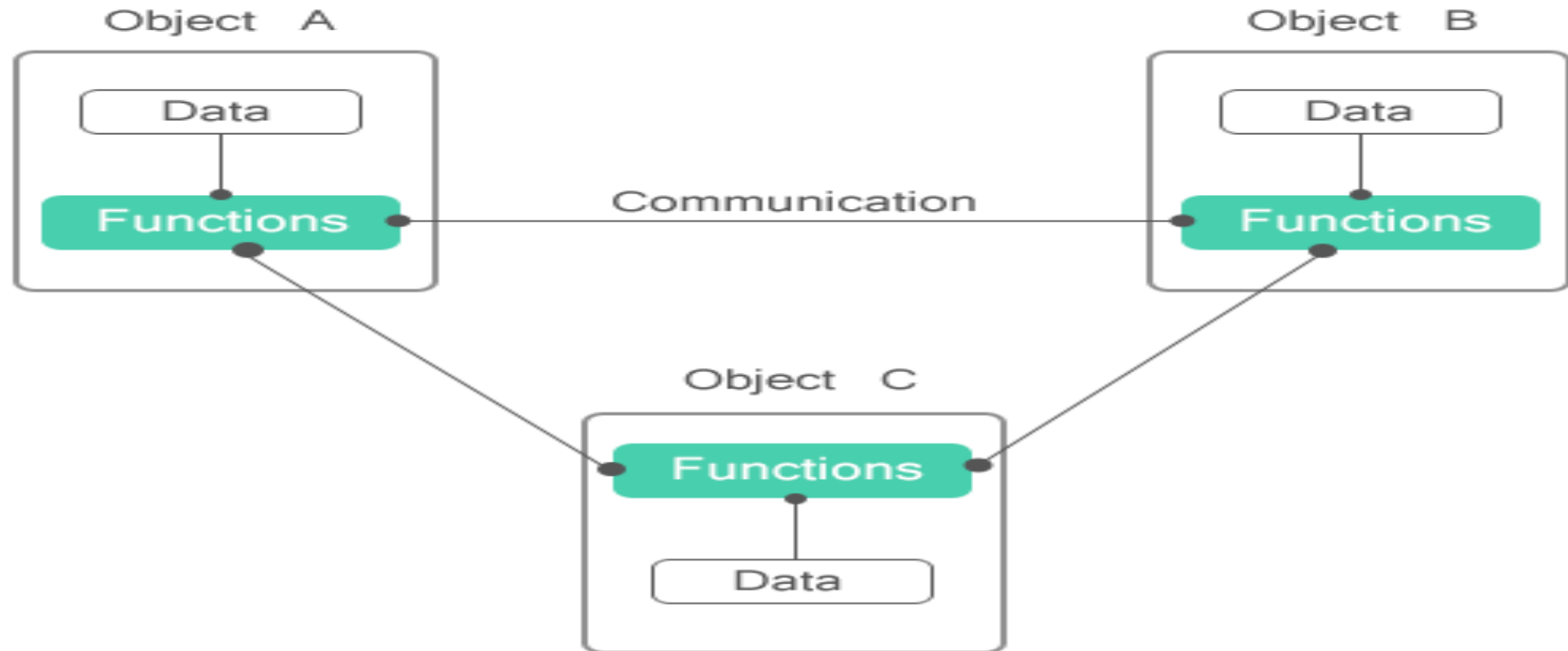
# Object-Oriented Approach

---

A programming paradigm called object-oriented programming (OOP), which is built on objects, and it tries to include the benefits of modularity and reusability.

Object oriented programming paradigm allows decomposition of the system into the number of entities called objects and then ties properties and function to these objects.

---



---

Object-oriented programming paradigm makes it easier to adapt to the changing requirements, easier to maintain, create modules of functionality, promote greater design, makes programme robust, and perform desired work efficiently.

# Advantages

---

1. Programs are divided into simple elements referred to as object
2. Programs organized around objects, grouped in classes
3. Data is hidden from external functions.
4. Functions operate on the properties of an object.
5. Follow the bottom-up approach in oop design

# References:

---

<http://www.differencebetween.info/difference-between-procedural-structural-and-object-oriented-programming-languages>



---

Thank You