

Name	Adrian Leo Pradanaa
NPM	2106718344

1. Programming paradigm refers to a style of programming. There are lots of programming languages that are well-known but all of them need to follow some strategy when they are implemented. And that strategy is a paradigm. In C it can be divide into two paradigms that is Imperative and Declarative:

Imperative Programming Language

- Procedural Programming Paradigm : Procedural programming allows splitting instructions into procedures.
- Parallel Processing Approach : Parallel processing is the processing of program instructions by dividing them among multiple processors.

Declarative Programming Language

- Logic Programming Paradigm : The logic programming paradigm isn't made up of instructions - rather it's made up of facts and clauses. It uses everything it knows and tries to come up with the world where all of those facts and clauses are true.
- Functional Programming : The key principle of this paradigm is the execution of a series of mathematical functions. All code is within a function. All variables are scoped to the function.

References:

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

2. Structured programming is a paradigm that aims to make programs easier to comprehend from a reader's point of view. It does this by linearizing the flow of control through a program. In structured programming, execution follows the writing order of the code. Structured programming caught favor with programming languages for its iconic opposition to the keyword **goto**, aiming to reduce the prevalence of spaghetti code. Some other controversial features that most languages have not adopted are avoiding early exit and opposition to exceptions for control flow.

Three flow of control in C :

- **Sequence Logic:**

Sequential logic as the name suggests follows a serial or sequential flow in which the flow depends on the series of instructions given to the computer. Unless new instructions are given, the modules are executed in the obvious sequence.

- **Selection Logic :**

Selection Logic simply involves a number of conditions or parameters which decides one out of several written modules. For example if-else and switch-case statement

- **Iteration Logic :**

In this, there requires a statement that initializes the condition controlling the loop, and there must also be a statement inside the module that will change this condition leading to the end of the loop. For example for, while and do-while.

References:

- <https://www.freetimelearning.com/c-language/c-language-control-structures-statements.php>

3. Because goto statement force a programs to jump from one part to another part of program, without any regard for what section of a program is active or in use loops can be broken this way, but in all but the direst circumstances, it should be avoided. This means that in order to maintain the coding standards, the programmer needs to deliberately choose a more complex method of achieving a goal ideally without the GOTO command.

References:

- <https://alvaro-videla.com/2015/02/programming-myths.html>

4. Selection control structures, conditional statements are features of a programming language which perform different computations or actions depending on whether a programmer-specified Boolean condition evaluates to true or false. There's two structure for selection control :
 - **If-Else** Control Structure : uses two way selection
 - **Switch-Case** : The case control structure is a multi-way selection. Case control structures compare a given value with specified constants and take action according to the first expression to match.

We can use selection control to determine whether you eligible for election for example :



```
If age > 17
    Output "You can vote."
False:
    Output "You can't vote."
End
```

References:

- <https://alvaro-videla.com/2015/02/programming-myths.html>

5. Iteration Control Structures, Iteration is the process where a set of instructions or statements is executed repeatedly for a specified number of time or until a condition is met. There's three structure for iteration control :

- **For and While** : which test the loop condition at the top of the loop the main difference is you have to initiate a specific number for For statement
- **Do-While** : the conditional expression appears at the end of the loop, so the statement(s) in the loop executes once before the condition is tested.

We can use iteration control for printing a word in a certain loop without write it manually for example :

```
count assigned zero

While count < 5

    Display "I love computers!"

    Increment count

End
```

References:

- <https://www.programiz.com/c-programming/c-do-while-loops>

