#computer #college-studies IT

By: Yashraj Maher

Syllabus of Data Structures

Unit 1: Data Structures and Algorithms

- **Data Structures**
  - Introduction to linear and non-linear data structures
- **Algorithm Analysis**
  - Growth rates
  - Estimating growth rates
  - Big O notation

Unit 2: Arrays

- **Need for Arrays**
- **Linear Arrays**
  - Representation of linear arrays:
    - Row measure order
    - Column measure order
- **Operations on Arrays**
  - Traversing
  - Insertion
  - Modification
  - Deletion

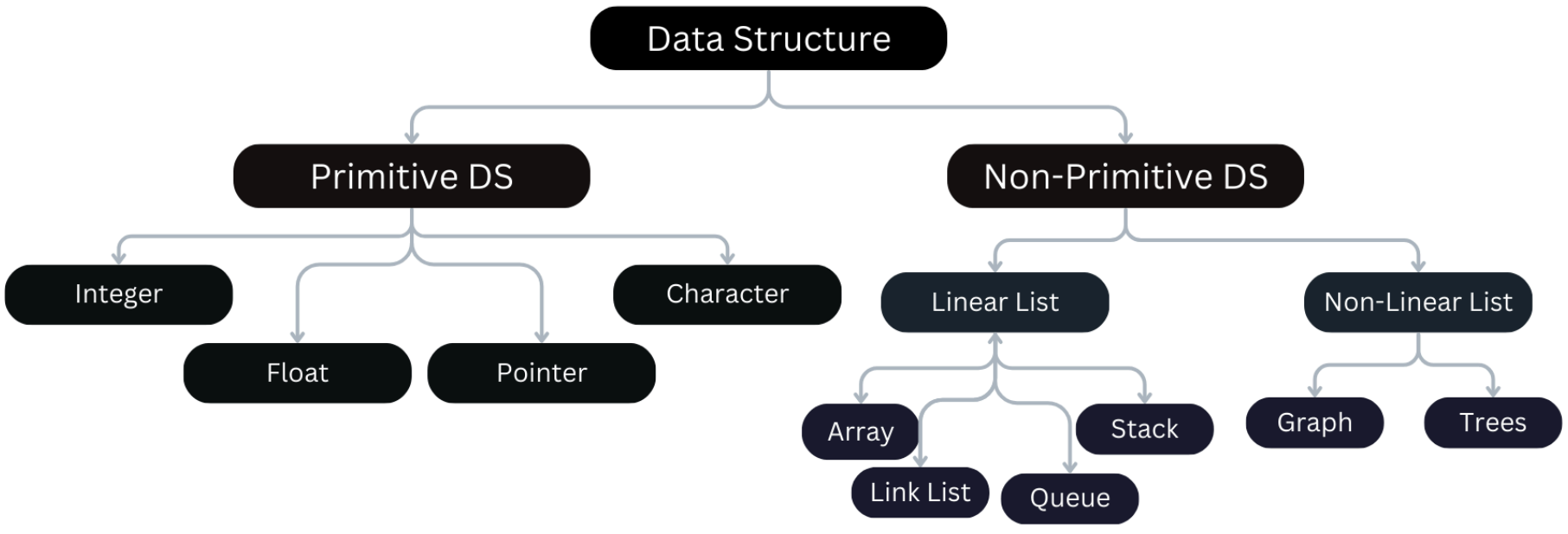
Data Structure Introduction

Definition:

- Data structure is representation of the logical relationship existing between individual elements of data.
- In other words, a data structure is a way of organizing all data items that considers not only the elements stored but also their relationship to each other.
- Data structure affects the design of both structure and functions aspects of a program.
  Program = algorithm + Data Structure
- You know that algorithm is step by step procedure to solve a particular function.
- That means algorithm is a set of instruction written to carry out certain task of the data structures is the way of organizing the data set with their logical relationship between.
- To develop a performance of an algorithm, we should use select an appropriate data structure for that algorithm.
- Therefore algorithm and its associated data structures from a programme.

Classification of Data Structures

- DS is broadly categorized in 2 types:
  - Primitive DS
  - Non-Primitive DS



Primitive Data Structure

- These are basic structures directly operated upon by the machine instructions.
- Two primal types are different representations on different languages.
- Integer, floating point number, character, constants, string constants, pointers, etc. fall in this category.

Non-Primitive Data Structures

- List, Stack, Queue, Tree, Graph are the examples of non-primitive data structures.
- The design of an efficient data structure must be done keeping in mind the operations to be performed on the data structure.
- The most commonly used operations for data structures are broadly categorized into following types:
  - Update
  - Selection
  - Updating
  - Searching
  - Sorting
  - Merging
  - Deleting or Editing

What is Fibonacci Series?

The Fibonacci series is a progression of numbers that begin with 0 & 1, in which subsequent numbers are calculated by adding the preceding two. This series is based on Fibonacci's problem where each number is equal to the sum of the two preceding numbers.

Example: Starting with 0 and 1, the sequence continues as follows:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ... etc.

---Here is the text from the images:

Algorithm of Fibonacci Series in C

To create an algorithm for generating the Fibonacci series in the C programming language, we will mind the following steps:

Steps:

1. Declare 3 variables A, B, C of type int (integer).
2. Set initial values A = 0 and B = 1.
3. Display the values for A & B.
4. Set C = A + B.
5. Display C.
6. Now set the value of A equal to the current value of B and B to the present value of C.
7. Repeat from step 4 for a desired number of elements.

Program:

```
# Include <stdio.h>

int main() {
    int Value1 = 0, Value2 = 1, Value3, Numbers;

    printf("Enter the number of terms: ");
    scanf("%d", &Numbers);

    printf("Fibonacci Series: ");

    for (int i = 1; i <= Numbers; ++i) {
        if (i == 1) {
            printf("%d, ", Value1);
            continue;
        }
        if (i == 2) {
            printf("%d, ", Value2);
            continue;
        }
        Value3 = Value1 + Value2;
        printf("%d, ", Value3);

        Value1 = Value2;
        Value2 = Value3;
    }

    return 0;
}
```