Data Structures & Algorithms - Overview

Data Structure is a systematic way to organize data in order to use it efficiently. Following terms are the foundation terms of a data structure.

- **Interface** − Each data structure has an interface. Interface represents the set of operations that a data structure supports. An interface only provides the list of supported operations, type of parameters they can accept and return type of these operations.

- **Implementation** − Implementation provides the internal representation of a data structure. Implementation also provides the definition of the algorithms used in the operations of the data structure.

Characteristics of a Data Structure

- **Correctness** − Data structure implementation should implement its interface correctly.

- **Time Complexity** − Running time or the execution time of operations of data structure must be as small as possible.

- **Space Complexity** − Memory usage of a data structure operation should be as little as possible.

Need for Data Structure

As applications are getting complex and data rich, there are three common problems that applications face now-a-days.

- **Data Search** − Consider an inventory of 1 million(10^6) items of a store. If the application is to search an item, it has to search an item in 1 million(10^6) items every time slowing down the search. As data grows, search will become slower.

- **Processor speed** − Processor speed although being very high, falls limited if the data grows to billion records.

- **Multiple requests** − As thousands of users can search data simultaneously on a web server, even the fast server fails while searching the data.

To solve the above-mentioned problems, data structures come to rescue. Data can be organized in a data structure in such a way that all items may not be required to be searched, and the required data can be searched almost instantly.

Execution Time Cases

There are three cases which are usually used to compare various data structure's execution time in a relative manner.

- **Worst Case** − This is the scenario where a particular data structure operation takes maximum time it can take. If an operation's worst case time is f(n) then this operation will not take more

than $f(n)$ time where $f(n)$ represents function of n.

- **Average Case** − This is the scenario depicting the average execution time of an operation of a data structure. If an operation takes $f(n)$ time in execution, then m operations will take m$f(n)$ time.

- **Best Case** − This is the scenario depicting the least possible execution time of an operation of a data structure. If an operation takes $f(n)$ time in execution, then the actual operation may take time as the random number which would be maximum as $f(n)$.

Basic Terminology

- **Data** − Data are values or set of values.

- **Data Item** − Data item refers to single unit of values.

- **Group Items** − Data items that are divided into sub items are called as Group Items.

- **Elementary Items** − Data items that cannot be divided are called as Elementary Items.

- **Attribute and Entity** − An entity is that which contains certain attributes or properties, which may be assigned values.

- **Entity Set** − Entities of similar attributes form an entity set.

- **Field** − Field is a single elementary unit of information representing an attribute of an entity.

- **Record** − Record is a collection of field values of a given entity.

- **File** − File is a collection of records of the entities in a given entity set.