

# Introduction & Overview

Unit - 1

(1)

## Basic Terminology: Elementary Data Organization

- Data: Are simply values or sets of values

- Data items: A single unit of values

- Group items: Data items that are divided into subitems are called group items.

- Elementary item: That are not included in group items.

- Entity:

- Entity set:

- Range: Each attribute of an entity set has a range of values, the set of all possible values that could be assigned to the particular attribute.

- Field

- Record

- File

- Primary key.

→ The above organization of data into fields, records & files may not be complex enough to maintain & efficiently process certain collections of data.

Study of Data structure, which forms the subject matter of & includes three steps:

① Logical or mathematical description of the structure.

② Implementation of the structure on a computer

③ Quantitative analysis of the structure, which includes determining the amount of memory needed to store the structure & time required to process the structure.

- There are situations where we want to delete the record with a given key, which <sup>may</sup> include first to search for the location of the record.

- The 2 operations, which are used in special situations are:

1. Sorting (Arranging the records in some logical order).
2. Merging (Combining the records in 2 different sorted files into a single sorted file).

## Data Type

- A data type defines a domain of allowed values & the operations that can be performed on those values.

- For example in C, int data type can take values in a range & operations that can be performed are addition, subtraction, multiplication, division, bitwise operations etc. Similarly in case of float, % operations, bitwise operations are not allowed.

- There are built-in data types or primitive data types & the values & operations for them are defined in language.

## Abstract data types (ADT)

- ADT is a mathematical model or concept that defines a data type logically.

- It specifies a set of data & collection of operations that can be performed on that data.

- The definition of ADT only mentions what operations are to be performed but not how these operations will be implemented.

- It does not specify how data will be organized in memory & what algorithm will be used for implementing the operations.

- It is called 'abstract' coz it gives an implementation independent view.

- The process of providing only the essentials & hiding the details is known as abstraction.

Some of the advantages of Data structure are!

1. Efficiency.
2. Reusability.
3. Abstraction.

## Linear & Non linear data Structures

### Linear DS

- 1) DS is linear if all the elements are arranged in a linear order.
- 2) In linear DS, each element has only one <sup>(1)</sup> successor & only one <sup>(1)</sup> predecessor.
- 3) The only exceptions are the first & the last elements, first element does not have a predecessor & last element does not have a successor.
- 4) Examples: Array, String, linked list, stack & queue.

### Non-linear DS

- There is no linear order in the arrangements of the elements.
- 2) Examples: Trees & graphs.

## Static & dynamic data Structures

### Static DS

- 1) The memory is allotted at compilation time only.
- 2) Therefore, mem. size is fixed & it can't be changed at run time.
- 3) Allows fast access to elements but insertion & deletion is expensive.
- 4) Example: Array.

### Dynamic DS

- 1) Memory is allocated at run time.
- 2) These DS has flexible size.
- 3) Allows fast insertion & deletion of elements but access to elements is slow.

→ Run time of algorithm means the time taken by prog. for execution.

→ Some data structure takes more space but improves the run time & some takes less space but it affects run time of algorithm.

→ The major criteria for complexity of any algorithm is comparison of keys & moving of data, means no. of times the key is compared & data is moved.

→ 3 cases for complexity of algorithm are :-

- (1) Best case (linear search)
- (2) Worst case.
- (3) Average case.

## 4) Example: Linked lists.

### Algorithm

- An algo is a procedure having well defined steps for solving a particular problem.
- The data stored in the DS is manipulated by using different algos, so study of DS includes study of algo.
- It makes the logic of program.
- Common approaches of algo. are:

### 1) Greedy Algorithm.

- It works by taking a decision that appears best at the moment, without thinking about the future.
- local optimum is chosen at every step in hope of getting a global optimum at the end.
- It is not necessary that a greedy algo will always give an optimal sol.
- Examples: Dijkstra's algo, Prim's & Kruskal's algo, & Huffman's algo.

### 2) Divide & Conquer Algo.

- Solves a problem by dividing it into smaller & similar subproblems.
  - The sol. of these smaller problems are then combined to get the sol. of the given problem.
  - Examples: Merge sort, Quick sort, Binary search.
- 3) Non recursive algorithm.
- 4) Randomized algorithm. — we use the feature of random no. instead of fixed no.
- 5) Modular programming approach → we divide the big problem into smaller ones which are totally different from each other.



## Data structures

- The logical or mathematical model of a particular organization of data is called a data structure.
- The choice of a particular data model depends on 2 points:
  1. It must be rich enough in structures to mirror the actual relationships of the data in the real world.
  2. The structure should be simple enough that one can effectively process the data when necessary.

- Some of the data structures which we will study are

1. Arrays [ 1D
  2. linked list [ 2D
  3. Trees.
  4. Stack
  5. Queue
  6. Graph
- data structures other than above 3.

## Data Structure Operations

The data appearing in our data structures are processed by some of the operations.

- Four operations are:
1. Traversing (Accessing each record exactly once so that some items in the record may be processed.)
  2. Searching (Finding the location of the record with a given key value or finding the locations of all records which satisfy one or more conditions).
  3. Inserting: (Adding a new record to the structure)
  4. Deleting: (Removing a record from the structure).