

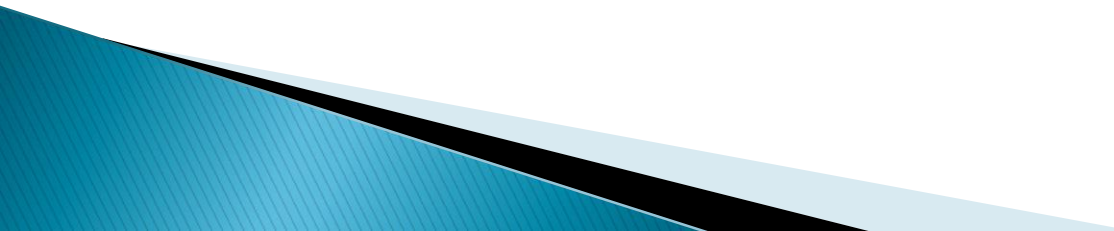
# Fundamentals of Data Structure and Algorithms



# Course Information


- ▶ Textbooks
- ▶ Fundamentals of data structure – Ellis Horowitz and Sartaj Sahni Chapters 2,3,4,5,6.1 to 6.3 (Excluding 2.3,3.2,4.5,4.6,5.8.3,5.9,4.7,4.10,4.12,6.4,6.5).
- ▶ Fundamental of Computer Algorithms – Ellis Horowitz and sartaj sahni Galgotia Publications.

# Course Outline

- ▶ Introduction to Data Structure
  - ▶ Arrays
  - ▶ Stacks
  - ▶ Queues
  - ▶ Lists and linked lists
  - ▶ Trees
  - ▶ Graphs
  - ▶ Algorithms
  - ▶ Divide and Conquer
  - ▶ Greedy Method
- 

# Introduction to Data Structure and Abstract Data Type(ADT)

# Introduction to Data Structure

- ▶ Computer science is the study of data, its representation and transformation by computer. For every data object, we consider the class of operations to be formed and then the way to represent the object so that these operations may be efficiently carried out. The two techniques required for this is
  - ▶ Devise alternative forms of data representation
  - ▶ Analyze the algorithm which operates on the structure.
  - ▶ These includes data structures, data types and data representations.
- 

## Data

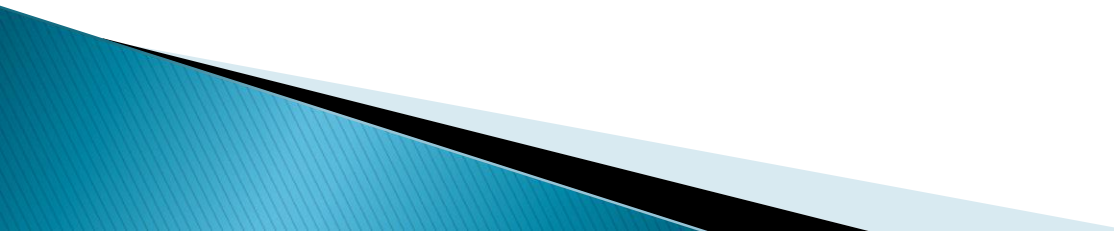
Data are simply values or set of values. A data items refers to a single unit of values. By data we mean known facts that can be recorded and that help implicit meaning.

## Data type and its necessity

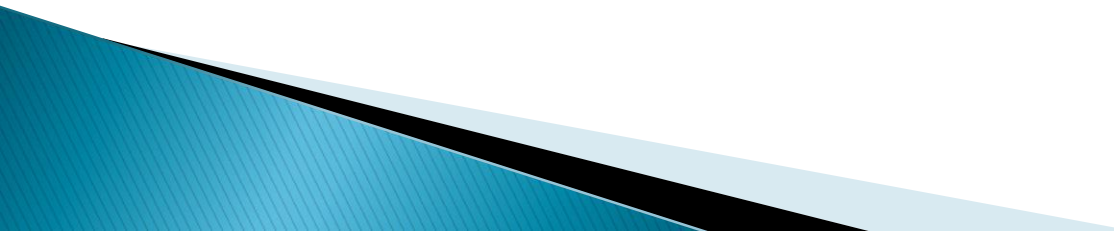
- ▶ A data type is a term which refers to the kind of data that a variable may hold. Every programming language has some built in data types. This means that the language allows variables to name data of that type and provides a set of operations which meaningfully manipulates these variables.

# Abstract Data Type and Data Structure

## ▶ Definition:–

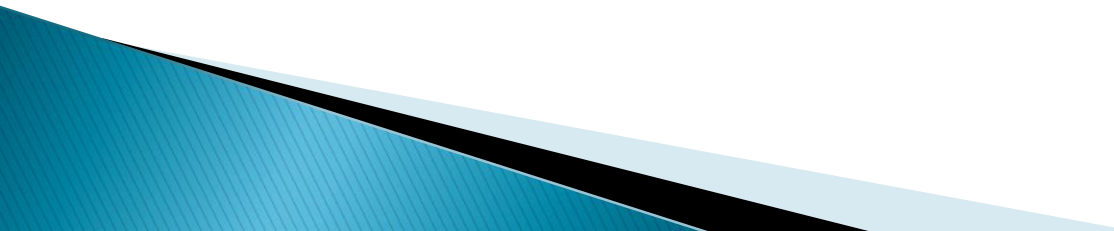
- Abstract Data Types (ADTs) stores data and allow various operations on the data to access and change it.
  - A mathematical model, together with various operations defined on the model
  - An ADT is a collection of data and associated operations for manipulating that data
- 

# Abstract Data Type


- ▶ ADTs support abstraction, encapsulation, and information hiding.
  - ▶ Abstraction is the structuring of a problem into well-defined entities by defining their data and operations.
  - ▶ The principle of hiding the used data structure and to only provide a well-defined interface is known as encapsulation.
- 



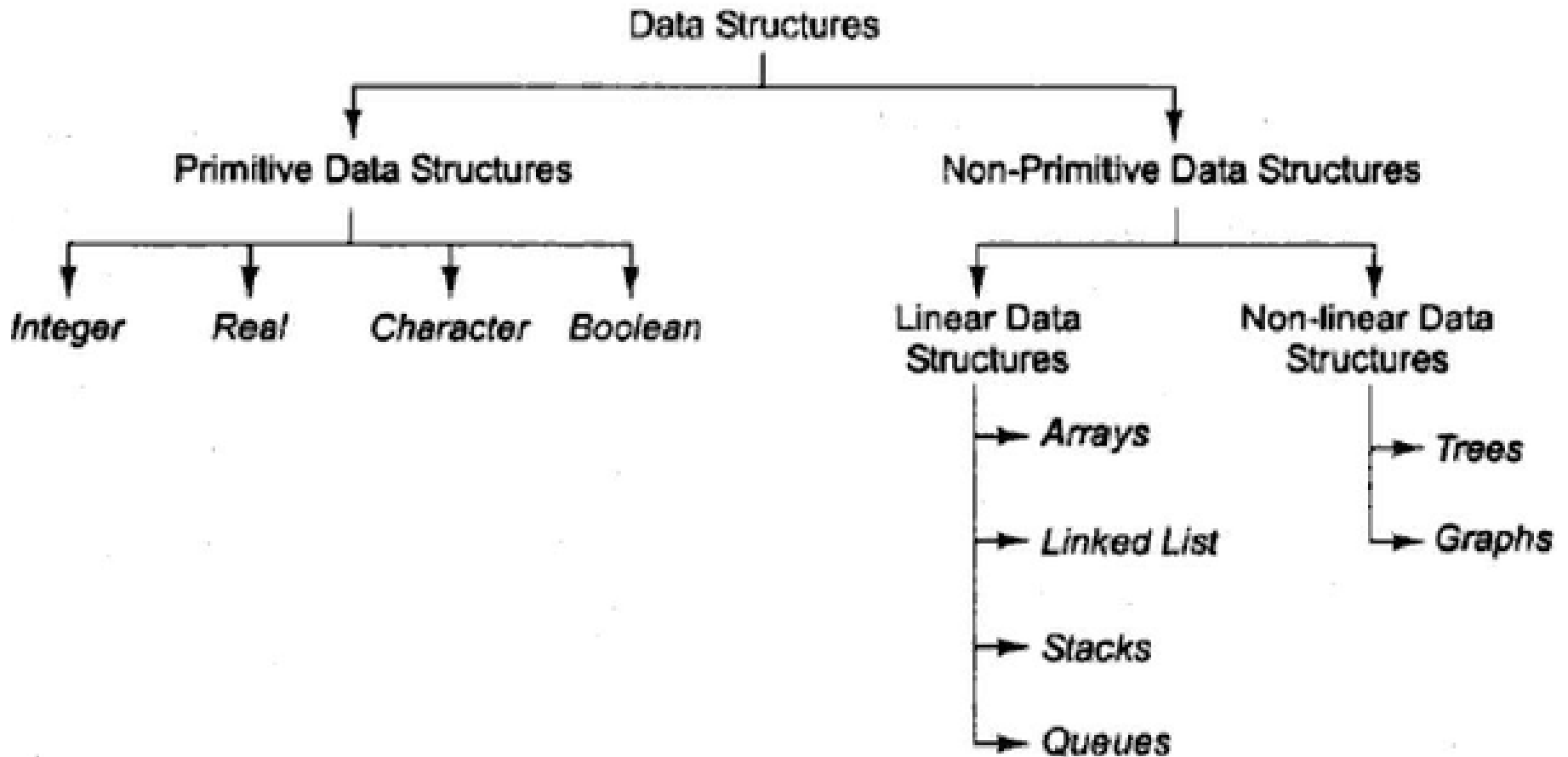
# The Core Operations of ADT

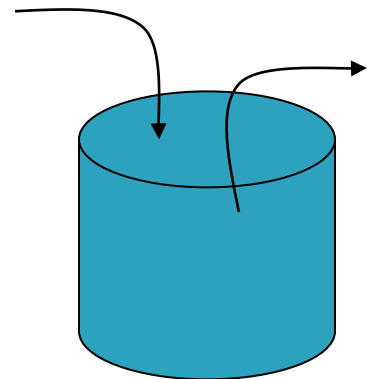
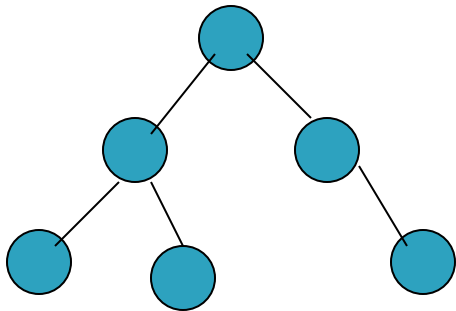
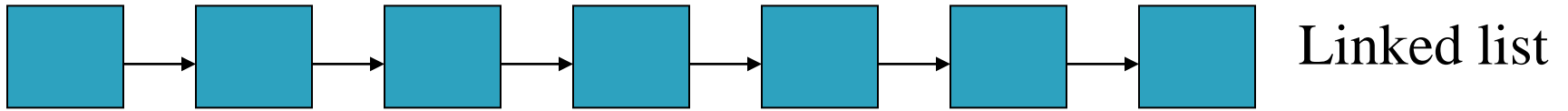
- ▶ Every Collection ADT should provide a way to:
    - add an item
    - remove an item
    - find,
    - Retrieve
    - access an item
- 

# What is Data Structure?

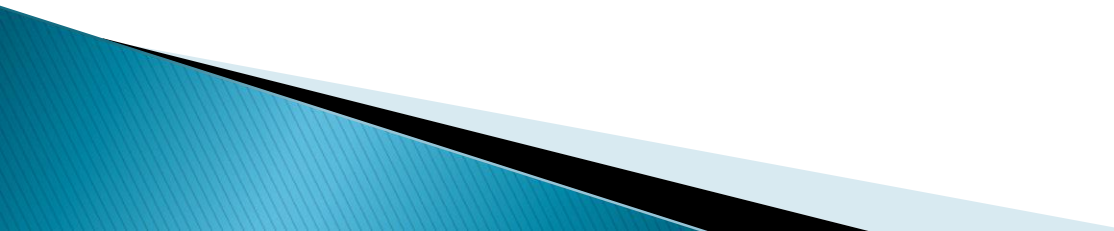
- ▶ The logical or mathematical model of a particular organization of data is called as data structure.
  - ▶ Data Structure that is implementing an ADT consists of variables for storing the data specified in the ADT and the algorithms for implementing the operations specified in ADT.
  - A data structure is a way to store and organize data in order to facilitate the access and modifications.
- 

# Types of Data Structures





# Selection of Data Structure

- ▶ The choice of particular data model depends on two considerations:
    - It must be rich enough in structure to represent the relationship between data elements
    - The structure should be simple enough that one can effectively process the data when necessary
- 

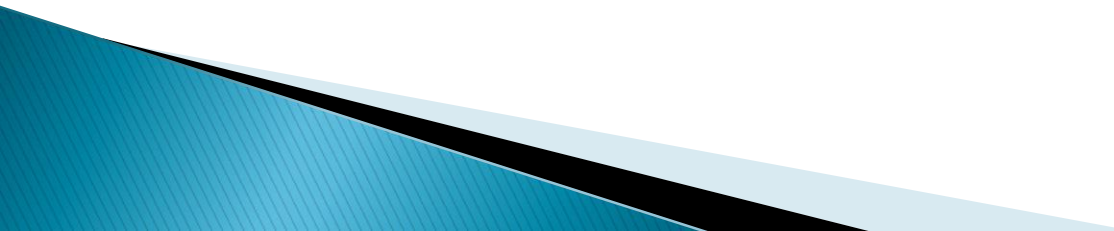
# Types of Data Structure

- ▶ Linear: In Linear data structure, values are arranged in linear fashion.
  - Array: Fixed-size
  - Linked-list: Variable-size
  - Stack: Add to top and remove from top
  - Queue: Add to back and remove from front
  - Priority queue: Add anywhere, remove the highest priority

# Types of Data Structure

- ▶ Non-Linear: The data values in this structure are not arranged in order.
  - Tree: Data is organized in branches.
  - Graph: A more general branching structure, with less strict connection conditions than for a tree

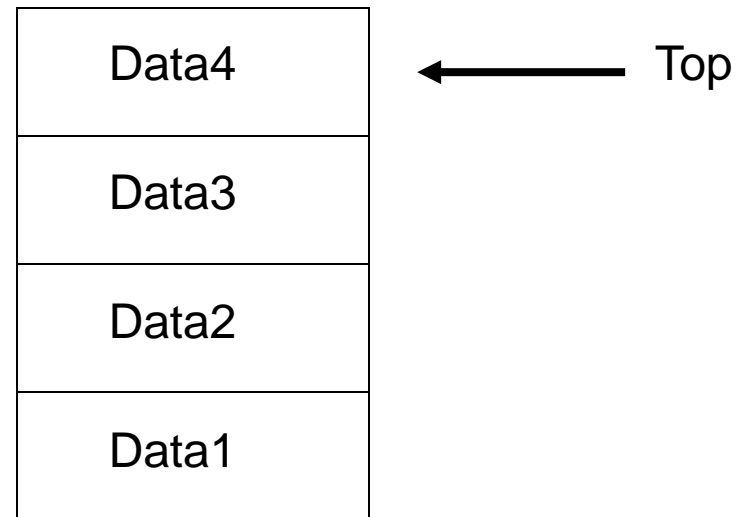
# Type of Data Structures

- ▶ Homogenous: In this type of data structures, values of the same types of data are stored.
    - Array
  - ▶ Non-Homogenous: In this type of data structures, data values of different types are grouped and stored.
    - Structures
    - Classes
- 



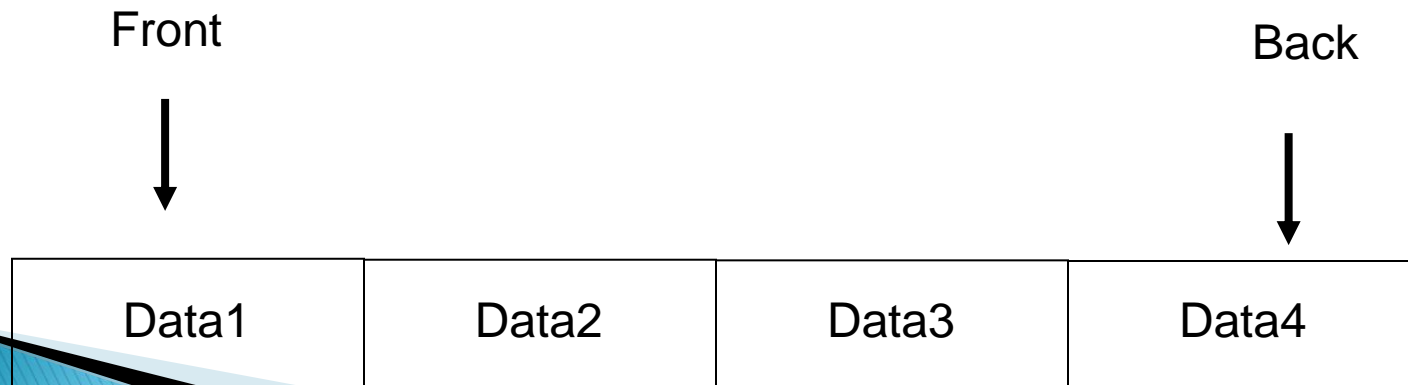
# Stacks

- ▶ Collection with access only to the last element inserted
- ▶ Last in first out
- ▶ insert/push
- ▶ remove/pop
- ▶ top
- ▶ make empty



# Queues

- ▶ Collection with access only to the item that has been present the longest
- ▶ Last in last out or first in first out
- ▶ enqueue, dequeue, front
- ▶ priority queues and dequeue



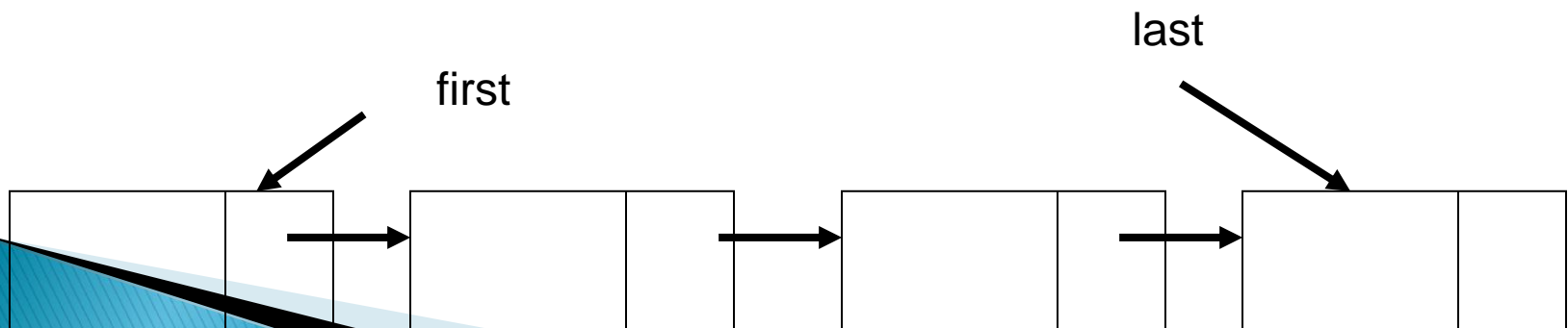
# List

- ▶ A *Flexible* structure, because can grow and shrink on demand.

Elements can be:

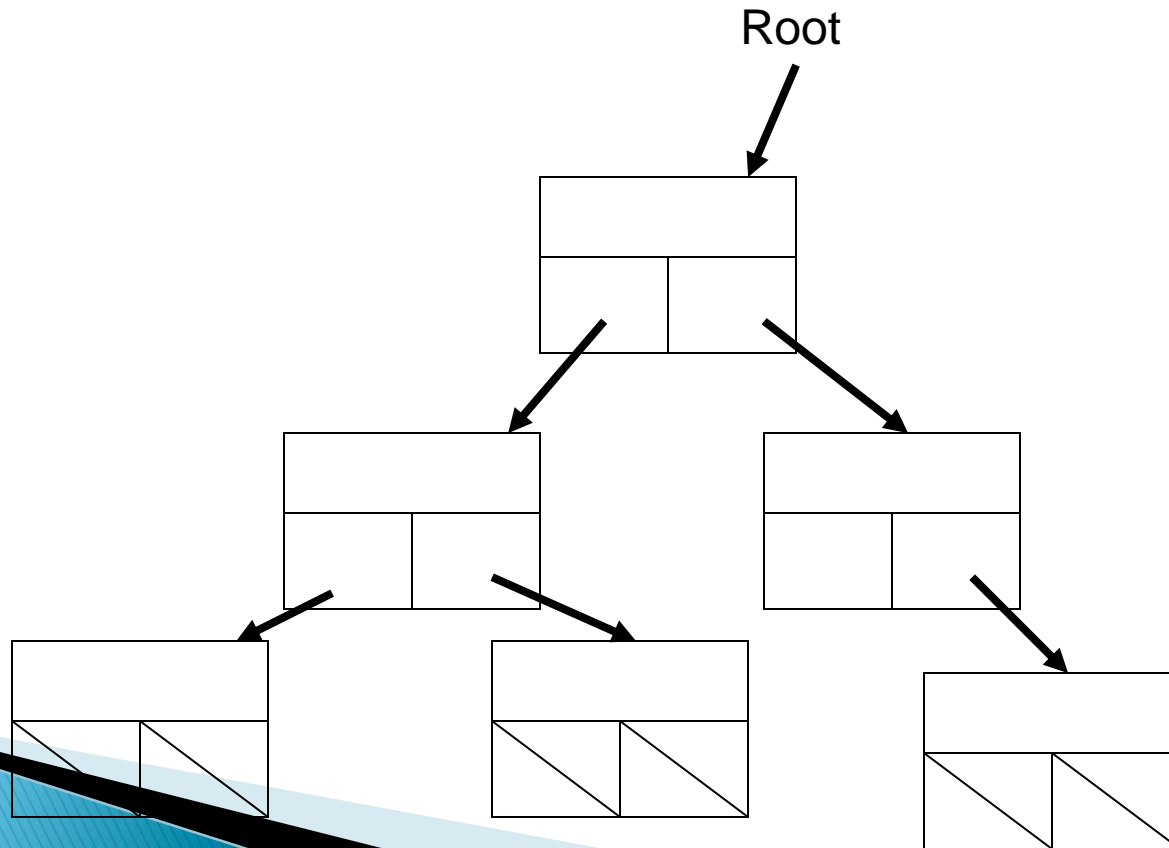
- Inserted
- Accessed
- Deleted

At *any* position



# Tree

- ▶ A **Tree** is a collection of elements called **nodes**.
- ▶ One of the node is distinguished as a **root**, along with a relation (“parenthood”) that places a hierarchical structure on the nodes.



# Graphs

- ▶ Data Structure that consists of set of nodes(vertices) and set of edges that relate the nodes to each other.
- ▶ Set of describe relationship among the vertices.

