



# **SNS COLLEGE OF TECHNOLOGY**

**Coimbatore-35  
An Autonomous Institution**

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade  
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai



## **DEPARTMENT OF CSE**

### **23ITT101-PROGRAMMING IN C AND DATA STRUCTURES**

**I YEAR - II SEM**

#### **UNIT 3 – ARRAYS AND INTRODUCTION TO DATA STRUCTURES**

##### **TOPIC 1 – Introduction to Arrays**



# INTRODUCTION TO ARRAY



- So far we have used only the fundamental data types, namely
  - **char, int, float, double** and variations of int and double.
- Although these types are very useful, they are constrained by the fact that a variable of these types can store **only one value** at any given time.
- Therefore, they can be used only to handle **limited amounts** of data.
- In many applications, however, we need to handle a large volume of data in terms of reading, processing and printing.
- To process such large amounts of data, we need a **powerful data type** that would facilitate efficient storing, accessing and manipulation of data items.
- C supports a derived data type known as **array** that can be used for such applications.



# INTRODUCTION TO ARRAY



- An array is a fixed-size sequenced collection of elements of the **same data type**.
- It is simply a **grouping of like-type data**.
- In its simplest form, an array can be used to represent a list of numbers, or a list of names.
- Some examples where the concept of an array can be used:
  - List of employees in an organization.
  - List of products and their cost sold by a store.
  - Test scores of a class of students.
  - Etc
- Since an array provides a convenient structure for representing data, it is classified as one of the **data structures** in C.
- Other data structures include structures, **lists, Stacks, queues and trees**.



# INTRODUCTION TO ARRAY



- An array is a **sequenced collection** of related data items that share a common name.
- For instance, we can use an array name **salary** to represent a set of salaries of a group of employees in an organization.
- We can refer to the individual salaries by writing a number called **index** or **subscript** in brackets after the array name.
- For example, **salary [10]** represents the salary of 10th employee.
- While the complete set of values is referred to as an array, individual values are called **elements**.
- The ability to use a single name to represent a collection of items and to refer to an item by specifying the item number enables us to develop concise and efficient programs.



# TYPES OF ARRAYS

➤ We can use arrays to represent not only **simple lists** of values but also **tables of data** in two, three or more **dimensions**.

➤ There are 3 types of arrays.

➤ One-dimensional arrays

➤ Two-dimensional arrays

➤ Multidimensional arrays

mark[0] mark[1] mark[2] mark[3] mark[4]

--	--	--	--	--

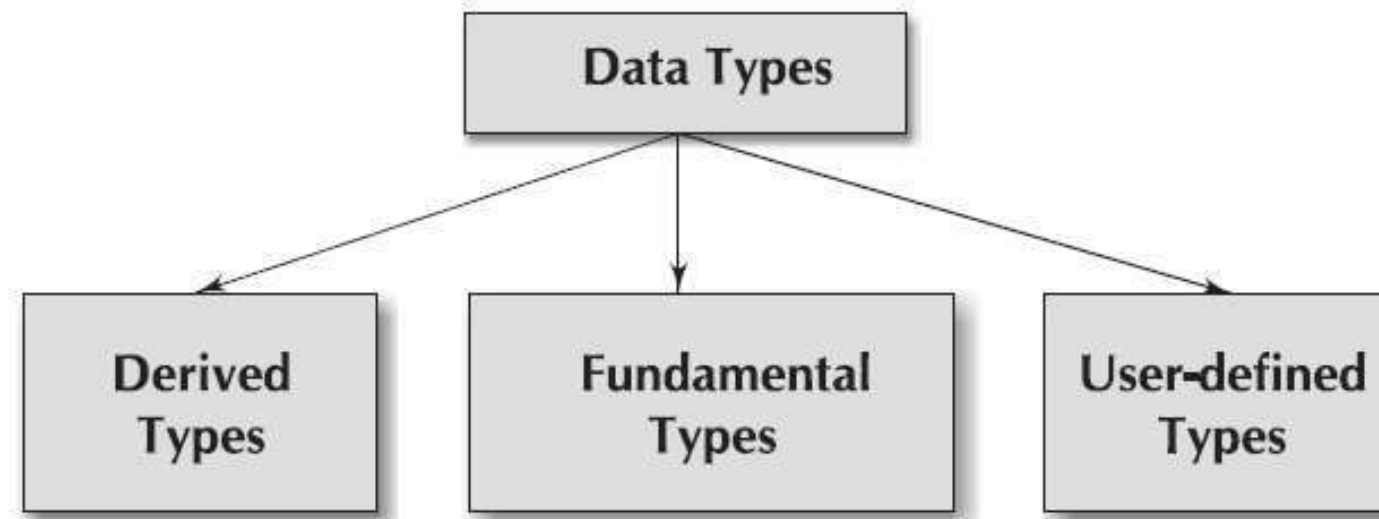
mark[0] mark[1] mark[2] mark[3] mark[4]

19	10	8	17	9
----	----	---	----	---



# DATA STRUCTURES

C supports a rich set of derived and user-defined data types in addition to a variety of fundamental types as shown below:



- Arrays
- Functions
- Pointers

- Integral Types
- Float Types
- Character Types

- Structures
- Unions
- Enumerations

- Arrays and structures are referred to as structured data types because they can be used to represent data values that have a structure of some sort.
- **Data Structures** are those with structured data types providing an organizational scheme that shows the relationships among the individual elements and facilitate efficient data manipulations.