MODULE 1: INTRODUCTION

DATA STRUCTURES

Data may be organized in many different ways. 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 the two considerations

1. It must be rich enough in structure 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 whenever necessary.

Basic Terminology: Elementary Data Organization:

Data: Data are simply values or sets of values.

Data items: Data items refers to a single unit of values.

Deepak D

Data items that are divided into sub-items are called **Group items**. Ex: An Employee Name may be divided into three subitems- first name, middle name, and last name.

Data items that are not able to divide into sub-items are called **Elementary items.**
Ex: SSN

Entity: An entity is something that has certain attributes or properties which may be assigned values. The values may be either numeric or non-numeric.

Ex:	Attributes-	Names,	Age,	Sex,	SSN
	Values-	Rohland Gail,	34,	F,	134-34-5533

Entities with similar attributes form an **entity set**. 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.
The term "information" is sometimes used for data with given attributes, of, in other words meaningful or processed data.

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

Record is the collection of field values of a given entity.

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

Scanned with CamScanner

Each record in a file may contain many field items but the value in a certain field may uniquely determine the record in the file. Such a field K is called a primary key and the values k1, k2, ... in such a field are called keys or key values.

Records may also be classified according to length.
A file can have fixed-length records or variable-length records.

- In fixed-length records, all the records contain the same data items with the same amount of space assigned to each data item.
- In variable-length records file records may contain different lengths.

Example: Student records have variable lengths, since different students take different numbers of courses. Variable-length records have a minimum and a maximum length.

The above organization of data into fields, records and files may not be complex enough to maintain and efficiently process certain collections of data. For this reason, data are also organized into more complex types of structures.

The study of complex data structures includes the following 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 and the time required to process the structure.

CLASSIFICATION OF DATA STRUCTURES

Data structures are generally classified into

- Primitive data Structures
- Non-primitive data Structures

1. **Primitive data Structures:** Primitive data structures are the fundamental data types which are supported by a programming language. Basic data types such as integer, real, character and Boolean are known as Primitive data Structures. These data types consists of characters that cannot be divided and hence they also called simple data types.

2. **Non- Primitive data Structures:** Non-primitive data structures are those data structures which are created using primitive data structures. Examples of non-primitive data structures is the processing of complex numbers, linked lists, stacks, trees, and graphs.

Scanned with CamScanner

Based on the structure and arrangement of data, non-primitive data structures is further classified into

1. Linear Data Structure
2. Non-linear Data Structure

1. Linear Data Structure:

A data structure is said to be linear if its elements form a sequence or a linear list. There are basically two ways of representing such linear structure in memory.

1. One way is to have the linear relationships between the elements represented by means of sequential memory location. These linear structures are called arrays.
2. The other way is to have the linear relationship between the elements represented by means of pointers or links. These linear structures are called linked lists.

The common examples of linear data structure are Arrays, Queues, Stacks, Linked lists

2. Non-linear Data Structure:

A data structure is said to be non-linear if the data are not arranged in sequence or a linear. The insertion and deletion of data is not possible in linear fashion. This structure is mainly used to represent data containing a hierarchical relationship between elements. Trees and graphs are the examples of non-linear data structure.

Deepak D

Arrays:

The simplest type of data structure is a linear (or one dimensional) array. A list of a finite number *n* of similar data referenced respectively by a set of *n* consecutive numbers, usually 1, 2, 3 . . . . . *n*. if **A** is chosen the name for the array, then the elements of **A** are denoted by subscript notation $a_1, a_2, a_3 \ldots . a_n$

or

by the parenthesis notation A (1), A (2), A (3) . . . . . A (n)

or

by the bracket notation A [1], A [2], A [3] . . . . . A [n]

Example 1: A linear array STUDENT consisting of the names of six students is pictured in below figure. Here STUDENT [1] denotes John Brown, STUDENT [2] denotes Sandra Gold, and so on.

	STUDENT
1	John Brown
2	Sandra Gold
3	Tom Jones
4	June Kelly
5	Mary Reed
6	Alan Smith

Scanned with CamScanner

Linear arrays are called one-dimensional arrays because each element in such an array is referenced by one subscript. A two-dimensional array is a collection of similar data elements where each element is referenced by two subscripts.

Example 2: A chain of 28 stores, each store having 4 departments, may list its weekly sales as in below fig. Such data can be stored in the computer using a two-dimensional array in which the first subscript denotes the store and the second subscript the department. If SALES is the name given to the array, then

SALES [1, 1] = 2872, SALES [1, 2] - 805, SALES [1, 3] = 3211,…., SALES [28, 4] = 982

Store \ Dept.	1	2	3	4
1	2872	805	3211	1560
2	2196	1223	2525	1744
3	3257	1017	3686	1951
. . .	. . .	. . .	. . .	. . .
28	2618	931	2333	982

Trees

Data frequently contain a hierarchical relationship between various elements. The data structure which reflects this relationship is called a rooted tree graph or a tree.

Some of the basic properties of tree are explained by means of examples

Example 1: Record Structure

Although a file may be maintained by means of one or more arrays a record, where one indicates both the group items and the elementary items, can best be described by means of a tree structure.

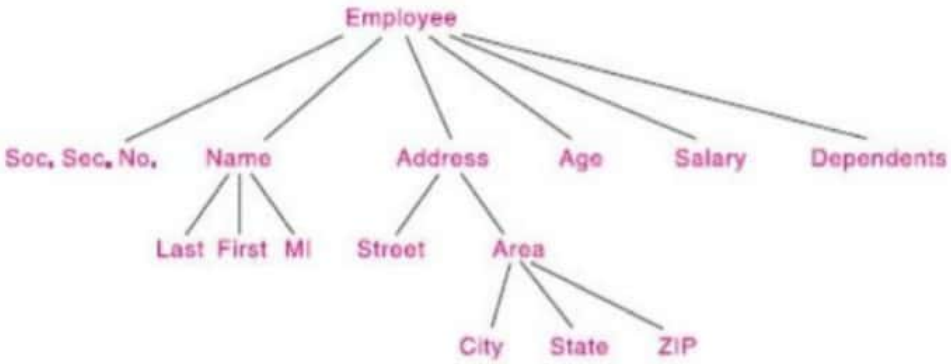
For example, an employee personnel record may contain the following data items:

Social Security Number, Name, Address, Age, Salary, Dependents

However, Name may be a group item with the sub-items Last, First and MI (middle initial). Also Address may be a group item with the subitems Street address and Area address, where Area itself may be a group item having subitems City, State and ZIP code number.

This hierarchical structure is pictured below

Scanned with CamScanner



(a)

Another way of picturing such a tree structure is in terms of levels, as shown below

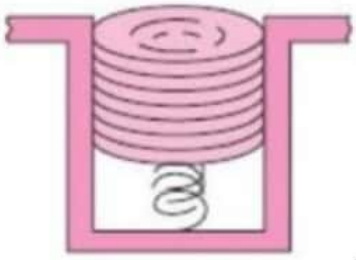
```
01 Employee
     02 Social Security Number
     02 Name
          03 Last
          03 First
          03 Middle Initial
     02 Address
          03 Street
          03 Area
               04 City
               04 State
               04 ZIP
     02 Age
     02 Salary
     02 Dependents
```

(b)

Some of the data structures are briefly described below.

1. Stack: A stack, also called a fast-in first-out (LIFO) system, is a linear list in which insertions and deletions can take place only at one end, called the top. This structure is similar in its operation to a stack of dishes on a spring system as shown in fig.

Note that new 4 dishes are inserted only at the top of the stack and dishes can be deleted only from the top of the Stack.



(a) Stack of dishes

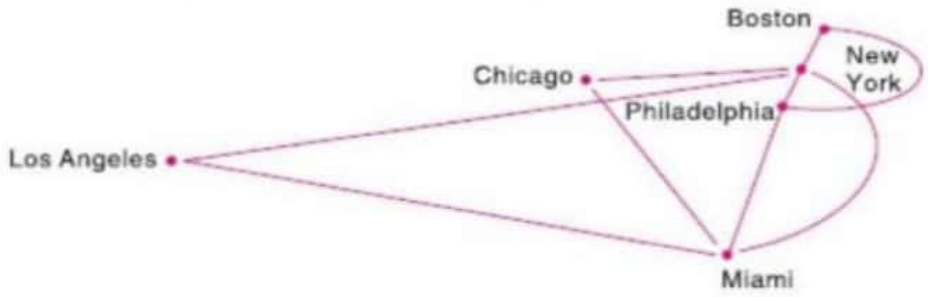
Scanned with CamScanner

2. Queue: A queue, also called a first-in first-out (FIFO) system, is a linear list in which deletions can take place only at one end of the list, the "from" of the list, and insertions can take place only at the other end of the list, the "rear" of the list.

This structure operates in much the same way as a line of people waiting at a bus stop, as pictured in Fig. the first person in line is the first person to board the bus. Another analogy is with automobiles waiting to pass through an intersection the first car in line is the first car through.

Queue waiting for a bus

3. Graph: Data sometimes contain a relationship between pairs of elements which is not necessarily hierarchical in nature. For example, suppose an airline flies only between the cities connected by lines in Fig. The data structure which reflects this type of relationship is called a graph.



Airline flights

Scanned with CamScanner

DATA STRUCTURES OPERATIONS

The data appearing in data structures are processed by means of certain operations.
The following four operations play a major role in this text:

1. **Traversing:** accessing each record/node exactly once so that certain items in the record may be processed. (This accessing and processing is sometimes called "**visiting**" the record.)
2. **Searching:** Finding the location of the desired node with a given key value, or finding the locations of all such nodes which satisfy one or more conditions.
3. **Inserting:** Adding a new node/record to the structure.
4. **Deleting:** Removing a node/record from the structure.

The following two operations, which are used in special situations:

1. **Sorting:** Arranging the records in some logical order (e.g., alphabetically according to some NAME key, or in numerical order according to some NUMBER key, such as social security number or account number)
2. **Merging:** Combining the records in two different sorted files into a single sorted file.

Scanned with CamScanner