ARRAYS

ARRAYS

- An array is a collection of similar data elements.
- The elements of the array are stored in consecutive memory locations and are referenced by an **index** (also known as the subscript).
- Declaring an array means specifying three things:
  - Data Type
  - Array Name
  - Array Size
- Arrays are declared using the following syntax.

ARRAYS

CALCULATING THE ADDRESS OF ARRAY ELEMENTS

Address of data element, A[k] = BA(A) + w(k – lower_bound)

Here, **A** is the array

k is the index of the element of which we have to calculate the address

BA is the base address of the array A.

w is the word size of one element in memory, for example, size of int is 2.

CALCULATING THE LENGTH OF THE ARRAY

Length = upper_bound – lower_bound + 1

Where, upper_bound is the index of the last element
and lower_bound is the index of the first element in the array

ARRAYS

Operations:

- **Traversing of Array**
- **Insertion in Arrays**
  - **Insert at end**
  - **Insert at front**
  - **Insert at a given position**
  - **Insert after a given value**
- **Deletion in Arrays**
  - **Delete from end**
  - **Delete from front**
  - **Delete at a given position**
  - **Delete a given value**
- **Linear Search**
- **Binary Search**

DELETE_LOC (Arr, SIZE, N, LOC)

Step 1. If N = = 0 then
PRINT "No data…No deletion"
End If
Step 2. If N > 0 AND LOC < N then
a. DELETE Arr [LOC]
b. Set I = LOC + 1
c. Repeat While I <= N-1 do
i. Set Arr [I-1] = Arr [I]
ii. Set I = I + 1
Done
d. DELETE Arr [N-1]
e. Set N = N – 1
Else
PRINT "LOC >= N, so no data to delete"
End If
Step 3. Exit