

PROGRAMMING CONCEPTS AND
EMBEDDED PROGRAMMING IN
C, C++ and JAVA:
Lesson-2: Data Structures: Arrays

Array

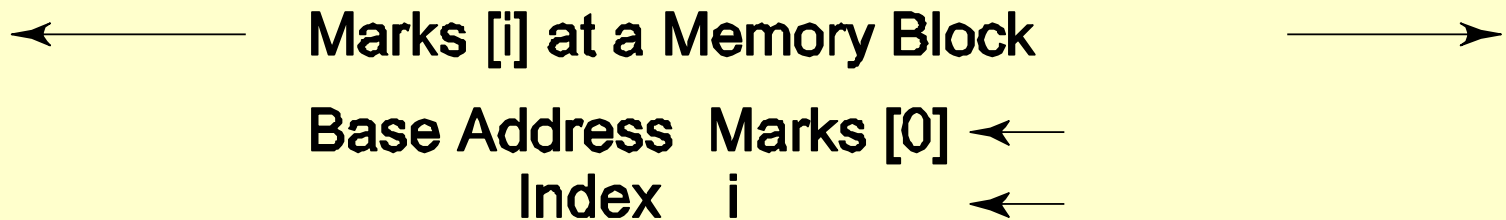
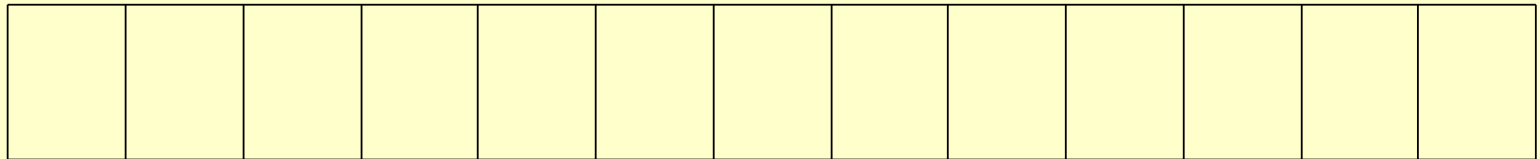
- Array: A structure with a series of data items sequentially placed in memory

Array

- (i) Each element accessible by an identifier name (which points to the array) and an index, i (which define offset from the first element)
- (ii) i starts from 0 and is +ve integer from 0 to $n - 1$ in case of n elements in array

An array at a memory block with one pointer for its base, first element with index = 0. Data word can be retrieved from any element by defining the pointer and index

Vector (One Dimensional Array)



One dimensional array (vector)

Example 1:

unsigned int *salary* [11];

salary [0] – 1st month salary

salary [11] – 12th month salary

Each integer is of 32-bit (4 bytes);

salary assigned 48 bytes address
space

One dimensional array (vector)

Example 2: *sio COM [1]*;

COM [0]– COM1 port data record with structure equivalent to *sio*

COM [1]– COM2 port data record with structure equivalent to *sio*

COM assigned $2*8$ characters = 16 bytes address space

Two dimensional array

Example 3:

`unsigned int salary [11, 9];`

`salary [3, 5]` – 4th month 6th year salary

`salary [11, 4]` – 12th month 5th year salary

`salary` assigned $12 * 10 * 4 = 480$ bytes address space

Multi-dimensional array

Example 4:

char pixel [143,175, 23];

pixel [0, 2, 5] – 1st horizontal line index *x*,
3rd vertical line index *y*, 6th color *c*.

pixel assigned $144*176*24 = 608256$
bytes address space in a colored
picture of resolution 144x 176 and 24
colors

Summary

We learnt

- **Array:** A structure with a series of data items sequentially placed in memory. Any data-element can be read or rewritten using the array pointer along with the element index(index) in the square bracket.

End of Lesson 2 of Chapter 7 on
Data Structures: Arrays