**Array**

**Main Points :**

* **What Is Array ?**
* **Advantages Of Array**

### **Disadvantages Of Array**

### **Time Complexity**

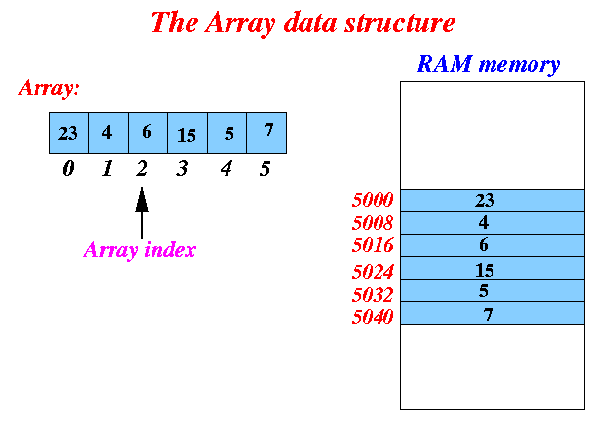
### **Space Complexity**

### **2D Array**

### **Resources**

**What Is Array ?**

* **collection of similar type of data items stored at contiguous memory locations.**

****

**Advantages Of Array :**

**1- Single Name , Multi Values**

**Instead Of Define Large Numbers Of Variables To Store Similer Data Type Items .. Array Has Single Name That Can Store Multi Values in it.**

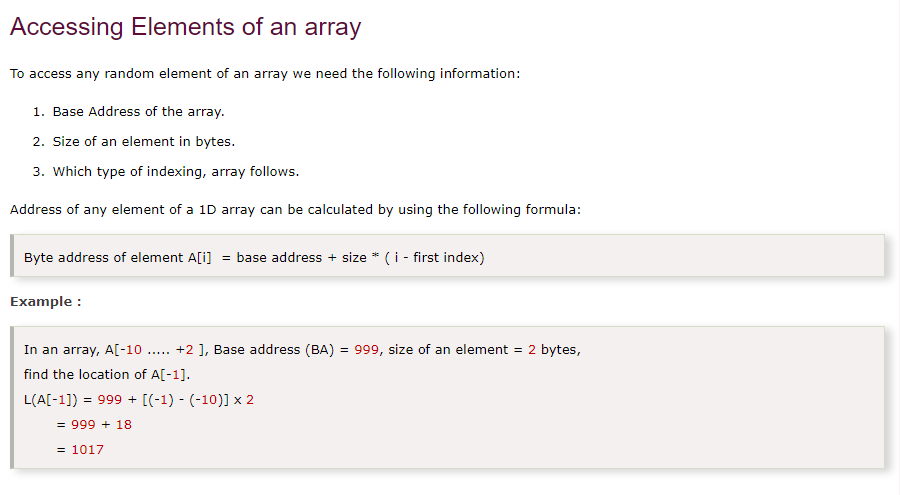
****

**2- Direct Access Using Index**

**Any element in any position in the array can be directly accessed using the index in just one step Because We Have Equation That Calculate Memory Address Of That Element.**

**Access Element Mean Read Or Write Value On It.**

**Access Element in Array --> Big(o) = 1**

****

**3- Array is a foundation of other data structures.**

**For example other data structures such as LinkedList, Stack, Queue etc. are implemented using array.**

### 

### **Disadvantages Of Array :**

**1- Fixed Size Or Static Structure :**

**1.1 We Need To Know The Size Of Array Before We Declare It**

**1.2 Empty Or Unused Elements Caused Waste In Memory**

If We Need Array Of 50 Elements And We Declare Array Of 100 Elements , So We Waste a lot Of Memory Size

**1.3 ReCreate New Array To Extend Old Array Size**

If We Need Array Of 50 Elements And Declare Array Of 50 Elements , But In The RunTime We Need To Extend The Size Of Array To 80 Elements , We Can Not Do That , And We Need To Declare New Array With Size Of 80 And Copy Old Elements In The New Array

**1.4 Insert New Element Is Very Slow : O(N)**

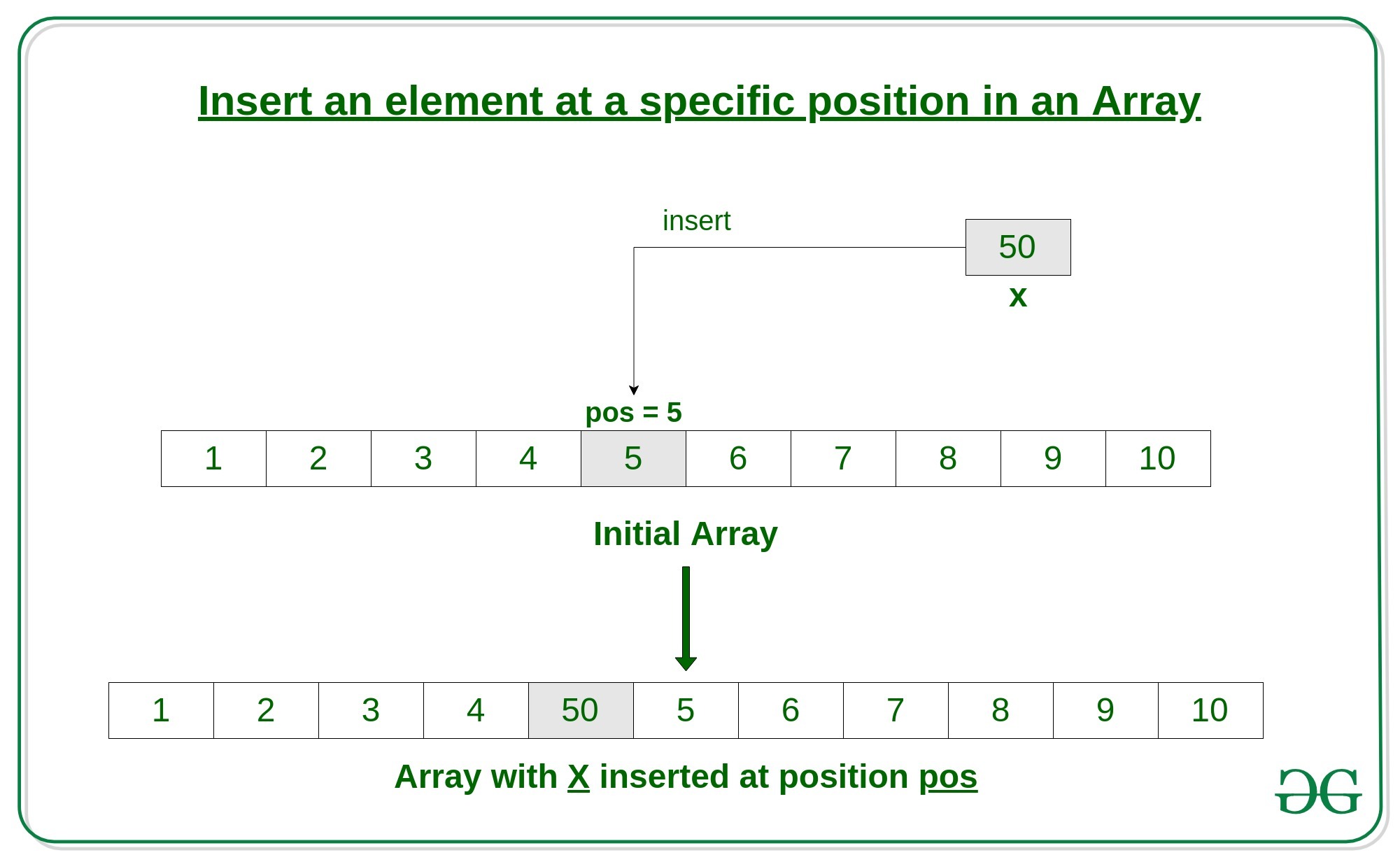
Actually We Can not Insert New Element Directly In Our Array Because it Has Static Structure.But We

1- Create New Array That Has Higher Size.

2- Copy Elements From Old Array To New Array Until We Reached To The Position Required.

3- Insert New Element At this Position.

4- Copy And Shift Remaining Items.



**1.5 Delete Element Is Very Slow : O(N)**

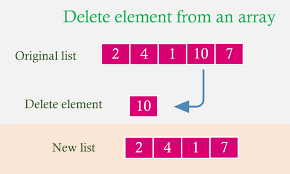
Actually We Can not Remove Element Directly From Our Array Because it Has Static Structure.But We

1- Create New Array With Lower Size.

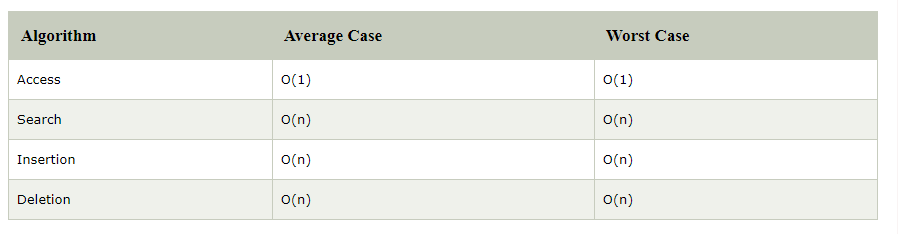
2- Copy Elements From Old Array To New Array Until We Reached To The Position Required.

3- Delete Element At this Position.

4- Copy And Shift Remaining Items.



**Time Complexity :**



### 

### **Space Complexity :**

In array, space complexity for worst case is **O(n)**.

### **2D Array :**

**- What is 2D array ?**

**- How to declare 2D Array ?**

**- How do we access data in a 2D array ?**

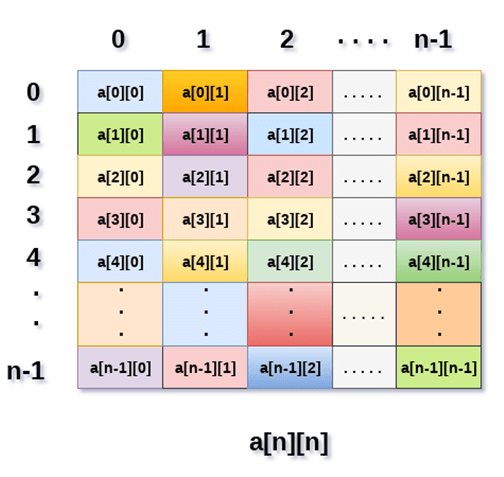
**- How To Store 2D Array in the memory ?**

**- Calculating Address of element of a 2D array ?**

### 

### **What is 2D Array ?**

* an array of arrays
* organized as matrices which can be represented as the collection of rows and columns.

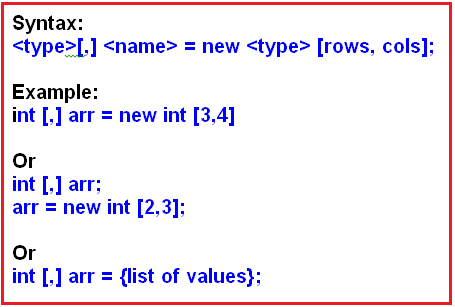


## 

## 

## **How to declare 2D Array ?**

* In C# :



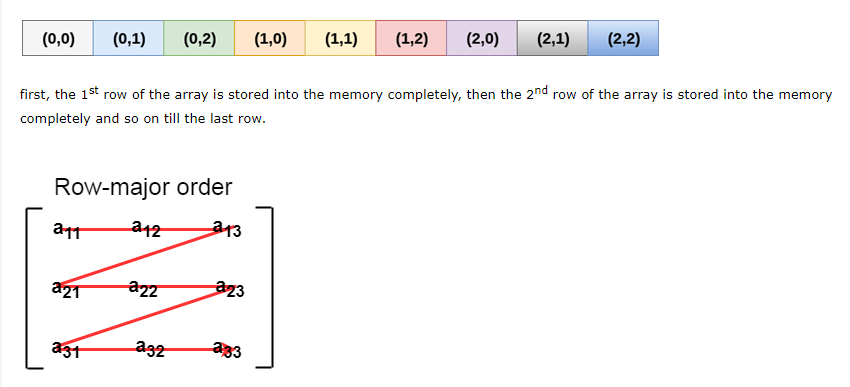
* **In C++ :**

****

### **How To Store 2D array into the memory**

There are two main techniques of storing 2D array elements into memory :

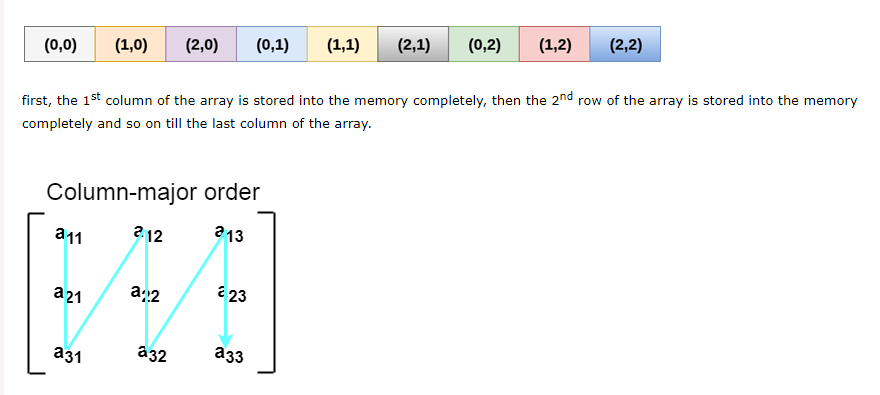
### **1. Row Major ordering ( Row By Row )**



### 

### 

### **2. Column Major ordering ( Column By Column )**



**Resources :**

**1-** [**https://www.javatpoint.com/data-structure-array**](https://www.javatpoint.com/data-structure-array)

**3-**

[**https://www.guru99.com/array-data-structure.html**](https://www.guru99.com/array-data-structure.html)

**4-** [**https://beginnersbook.com/2018/10/data-structure-array/**](https://beginnersbook.com/2018/10/data-structure-array/)

**5-**

[**http://www.mathcs.emory.edu/~cheung/Courses/170/Syllabus/09/basics.html**](http://www.mathcs.emory.edu/~cheung/Courses/170/Syllabus/09/basics.html)

**6-**

[**https://career.guru99.com/top-50-array-interview-questions-answers/**](https://career.guru99.com/top-50-array-interview-questions-answers/)

**7-**

[**https://hackernoon.com/50-data-structure-and-algorithms-interview-questions-for-programmers-b4b1ac61f5b0**](https://hackernoon.com/50-data-structure-and-algorithms-interview-questions-for-programmers-b4b1ac61f5b0)