Arrays - DSA by Coding Age

What is Data Structure?

A data structure is a specialized format for organizing, processing, retrieving and storing data. There are several basic and advanced types of data structures, all designed to arrange data to suit a specific purpose. Data structures make it easy for users to access and work with the data they need in appropriate ways. Most importantly, data structures frame the organization of information so that machines and humans can better understand it.

In computer science and computer programming, a data structure may be selected or designed to store data for the purpose of using it with various algorithms. In some cases, the algorithm's basic operations are tightly coupled to the data structure's design. Each data structure contains information about the data values, relationships between the data and -- in some cases -- functions that can be applied to the data.

For instance, in an object-oriented programming language, the data structure and its associated methods are bound together as part of a class definition. In non-object-oriented languages, there may be functions defined to work with the data structure, but they are not technically part of the data structure.

Typical base data types, such as integers or floating-point values, that are available in most computer programming languages are generally insufficient to capture the logical intent for data processing and use. Yet applications that ingest, manipulate and produce information must understand how data should be organized to simplify processing. Data structures bring together the data elements in a logical way and facilitate the effective use, persistence and sharing of data. They provide a formal model that describes the way the data elements are organized.

Why are data structures important?

Data structures are the building blocks for more sophisticated applications. They are designed by composing data elements into a logical unit representing an abstract data type that has relevance to the algorithm or application. An example of an abstract data type is a "customer name" that is composed of the character strings for "first name," "middle name" and "last name."

It is not only important to use data structures, but it is also important to choose the proper data structure for each task. Choosing an ill-suited data structure could result in slow runtimes or unresponsive code.

> Five factors to consider when picking a data structure include the following:

- What kind of information will be stored?
- How will that information be used?
- Where should data persist, or be kept, after it is created?
- What is the best way to organize the data?
- What aspects of memory and storage reservation management should be considered?

ALGORITHM

An **algorithm** is a set of steps for accomplishing a task or solving a problem. Typically, algorithms are executed by computers, but we also rely on algorithms in our daily lives. Each time we follow a particular step-by-step process, like making coffee in the morning or tying our shoelaces, we are in fact following an algorithm.

In the context of **computer science**, an algorithm is a mathematical process for solving a problem using a finite number of steps. Algorithms are a key component of any computer program and are the driving force behind various systems and applications, such as navigation systems, search engines, and music streaming services.

Linear Data Structure

Data structure in which data elements are arranged sequentially or linearly.

- Array
- Stack
- Queue
- Linked List

Static data structure: Static data structure has a fixed memory size.

- Array

Dynamic data structure: In dynamic data structure, the size is not fixed.

- Vector
- Linked List

Non-linear data structure

Data structures where data elements are not placed sequentially or linearly

- Tree
- Graph

Let’s deep dive into Arrays now.

1D ARRAYS

What is an Array?

- An array is a collection of similar data elements stored at continues memory locations.it stores homogeneous data.
- It is the simplest data structure where each data element can be accessed directly by only using its index number.

How to declare an array?

```
int arr[5] ;
```

Memory Allocation

- `int` - datatype.
- `arr` - array name.
- `5` - arraysize.
- arr ☐ ☐ ☐ ☐ ☐

  0 1 2 3 4 (index num.)

```
int  size=5;
```

```
int arr[size];
```

How to initialize an array?

```
int arr[3]={4,2,3};
```

• Memory Allocation

- arr [4] [2] [3]
- 0 1 2
- (Index num.)

Declare and initialize together.

`int arr[ ]={5,6,2};`

How to find the size of an array?

`int size=sizeof(Array name)/sizeof(data type);`

```
int arr[ ]={5,6,2};

int size=sizeof(arr)/sizeof(int);

printf("size=%d",size);  **output**

****size=**4**
```

Or

```
int size=sizeof(arr)/sizeof(arr[0]);
printf("size=%d",size);

**output**

****size=**4**
```

How to print Array elements?

```
int arr[3]={5,3,9};
printf("%d ",arr[0]);
printf("%d ",arr[1]);
printf("%d ",arr[2]);

- Output
- 5 3 9
```

How to take input from user in Array?

```
int size=5;
int arr[size];

for (int i = 0; i < size; i++) {
    scanf("%d", &a[i]);
}
```

How to print Array elements ?

```
for (int i = 0; i < size; i++) {
    printf("%d   ", a[i]);
}
```

1D ARRAY QUESTIONS :-

1. WAP to take input from user and print using Array ?
2. WAP to take input in array using function and print element using function ?
3. WAP to take input from user in array and find the sum of all elements?

`input - [3,5,2,1]`

`output - sum=11`

4. WAP to take input from user in array and find the multiply of all elements?

`input  [2,5,3,1]`

`output - multi=30.`

5. **WAP** to take input from user in array and print all even elements?

`input - [6,5,3,4,8]`

`output - [6,4,8]`

6. WAP to take input from user in array and find average of all odd element?

`input - [2,5,3,8]`

`output - Avg. of odd element=4.`

7. WAP to take input from user in array and display them in reverse order?

```
input - [2,6,3,9]
output -  [ 9, 3 , 6, 2 ]
```

8. WAP to merge two arrays in single array?

- ```
  input
  arr1 [2,5,3,8]
  arr2 [1,6,4,9]
  ```
- ```
  output
  arr3[2,5,3,8,1,6,4,9]
  ```

9. WAP to find maximum element in array?

- ```
  input - [1,6,7,9]
  ```
- ```
  output -  max=9
  ```

10.WAP to find minimum element in array?

- ```
  input -   [2,4,3,5]
  ```
- ```
  output - min=2
  ```

11.WAP to take input from user in array and the element is only **0** or **1** or other number is invalid?

```
input
Size=4

1
0
3 invalid
1
6 invalid
1
```

12.WAP to find 2nd maximum element in array?

- ```
  input - [3,5,1,9]
  output-  max=5
  ```

13.WAP to find 2nd minimum element in array?

- ```
  input - [2,6,3,9]
  ```

- `output - min =3`

14.WAP to find **k**th element in array?

- `input - [4,8,6,9] •k=2.`
  `output - Kth element=8`

15.WAP to take input in array and arrange in ascending order?

- `input - [7,3,6,2]`
  `output - [ 2,3,6,7 ]`

16. WAP to take input in array and arrange in descending order?

- `input - [9,3,5,2]`
  `output - [ 9,5,3,2 ]`

17.WAP to find **k**th maximum element in array?

- `input - [6,4,8,9] , k=2.`
  `output - 2nd max element=8`

18.WAP to find **k**th minimum element in array?

- `input - [7,4,2,9] •k=3.`
  `output - 3rd min element=7`

19.WAP to take input in array and delete **k**th element in array?

- `input - [3,4,8,2] •k=2.`

  `output [ 3,8,2 ]`

20.WAP to reverse an array?

- `input - [7,2,8,9]`
  `output - [ 9,8,2,7 ]`

21.WAP in c to separate odd and even integers into separate arrays.

`i/p =[2,5,11,6,44,31]`

`o/p =`

```
The Even elements are :
[2,6,44]
The Odd elements are :
[5,11,31]
```

22. Write a program to find the first repeating element in an array?

```
input:[3,2,4,7,2,9,5,1,2 ]

output: 2 is repeated, count =3
```

23.WAP to take input in array and remove all duplicates numbers?

```
input - [ 3,2,3,7,2,9,3,1,2,7 ]

output - [ 3,2,7,9,1 ]
```

24.WAP to take input in array and print all duplicates numbers and its count?

```
I nput [ 3,2,3,7,2,9,3,1,2,7 ]
Output -
-3 duplicate.
Count=3.
- 2 duplicate.
Count=3.
- 7 duplicate.
Count=2.
```

25.WAP to take an array n size and take A number from user and check number that number is present or not present in the array?

- `[ 4,21,76,1,9] •k= 76 - 76 is present.`
- `[8,32,54,65] •k=25 - 25 is not present.`

26.WAP in c to find a pair with given sum in the array?

```
input
```

- `[6,8,4,-5,7,9],k=15`

```
output
```

- `Value of 0 or 5 index sum is equals to 15.`
- `Value of 1 or 4 index sum is equals to 15.`

```
input
```

- `[2,8,4,-5,3,1],k=3`

```
output
```

- `Value of 0 or 5 index sum is equals to 3.`
- `Value of 1 or 3 index sum is equal to 3.`

27. Write a program to cyclically rotate an array by one?

```
Input : { 1, 2, 3, 4, 5 }
Output : { 5, 1, 2, 3, 4 }
```

28. Write a program to rotate an array by d elements?

```
Input : { 1, 2, 3, 4, 5, 6, 7 }, d = 2
Output : {6,7,1,2, 3, 4, 5 }
```

29. Write a program to find the missing number in an array of consecutive numbers

```
Input : { 1, 2, 3, 5, 6, 7, 8 }
Output: The missing number is 4.
```

30. Write a program to find the union and intersection of two sorted arrays?

```
Input : Array1 = { 1, 2, 4, 5, 6 }, Array2 = { 2, 3, 5, 7 }
Output : Union = { 1, 2, 3, 4, 5, 6, 7 }, Intersection = { 2, 5 }
```

31. Write a program to find the maximum sum subarray in an array?

```
Input : { -2, -3, 4, -1, -2, 1, 5, -3 }
Output : The maximum sum subarray is { 4, -1, -2, 1, 5 }
```

32. Write a program to find the leaders in an array?

```
Input : { 16, 17, 4, 3, 5, 2 }
Output : The leaders are 17, 5, and 2
```

33. Wap to find mountain in array?

```
Input : { 16, 17, 11, 3, 5, 2, 8 }
Output : 17 and 5 is mountain.
```

34. Write a program to sort an array in wave form

```
Input : { 10, 5, 6, 3, 2, 20, 100, 80 }
Output : { 10, 5, 6, 2, 20, 3, 100, 80 }
```

35. Write a program to rearrange positive and negative numbers in an array

```
Input : { -1, 2, -3, 4, 5, 6, -7, 8, 9 }
Output : { 2, 4, 5, 6, 8, 9, -1, -3, -7 }
```

36. Write a program to find the maximum difference between two elements in an array

```
Input : { 2, 3, 10, 6, 4, 8, 1 }
Output : The maximum difference is 9
```

37. Write a program to find the common elements in three sorted arrays?

```
Input : Array1 = { 1, 5, 10, 20, 40, 80 }, Array2 = { 6, 7, 20, 80, 100 }, Array3 = { 3, 4, 15, 20, 30, 70, 80, 120 }
Output : Common elements are 20 and 80
```

38. Write a program to count pairs in an array whose sum is divisible by "k"?

```
Input : { 2, 2, 1, 7, 5, 3 }, k = 4
Output : The count of pairs is 5
```

39. Write a program to rearrange an array in maximum minimum form?

```
input : { 1, 2, 3, 4, 5, 6, 7 }
Output : { 7, 1, 6, 2, 5, 3, 4 }
```

40. Write a program to find the element that appears once in an array where every other element appears twice?

```
Input : { 7, 3, 5, 4, 5, 3, 4 }
Output : The element that appears once is 7
```

41. Write a program to segregate 0s and 1s in an array?

```
Input : { 0, 1, 0, 1, 1, 1 }
Output : { 0, 0, 1, 1, 1, 1 }
```

42. Write a program to sort an array of 0s, 1s and 2s?

```
Input : { 0, 1, 2, 0, 1, 2 }
Output : { 0, 0, 1, 1, 2, 2 }
```

43. Write a program to print all the subsets of an array?

```
Input : { 1, 2, 3 }
Output: { {},{1},{2},{3},{1,2},{1,3},{2,3},{1,2,3} }
```

44. Write a program to find the next greater element for each element in an array?

```
Input : { 4, 5, 2, 25 }
Output : Next greater element for each element in the array is { 5, 25, 25, -1 }
```

45. Write a program to rearrange an array such that arr[i] = i ?

```
Input : { -1, -1, 6, 1, 9, 3, 2, -1, 4, -1 }
Output : { -1, 1, 2, 3, 4, -1, 6, -1, -1, 9 }
```

2D ARRAYS

What is two-dimensional arrays?

- A 2D array is also known as a matrix .
- The 2d array is an array that is organized in rows and columns. We can identify each element in a 2d array using the position of rows and columns.
- An array of arrays is a collection of similar data elements stored at continues memory locations.it stores homogeneous data.

How to declare an 2d array?

SYNTAX -

```
int arr[2][3];
```

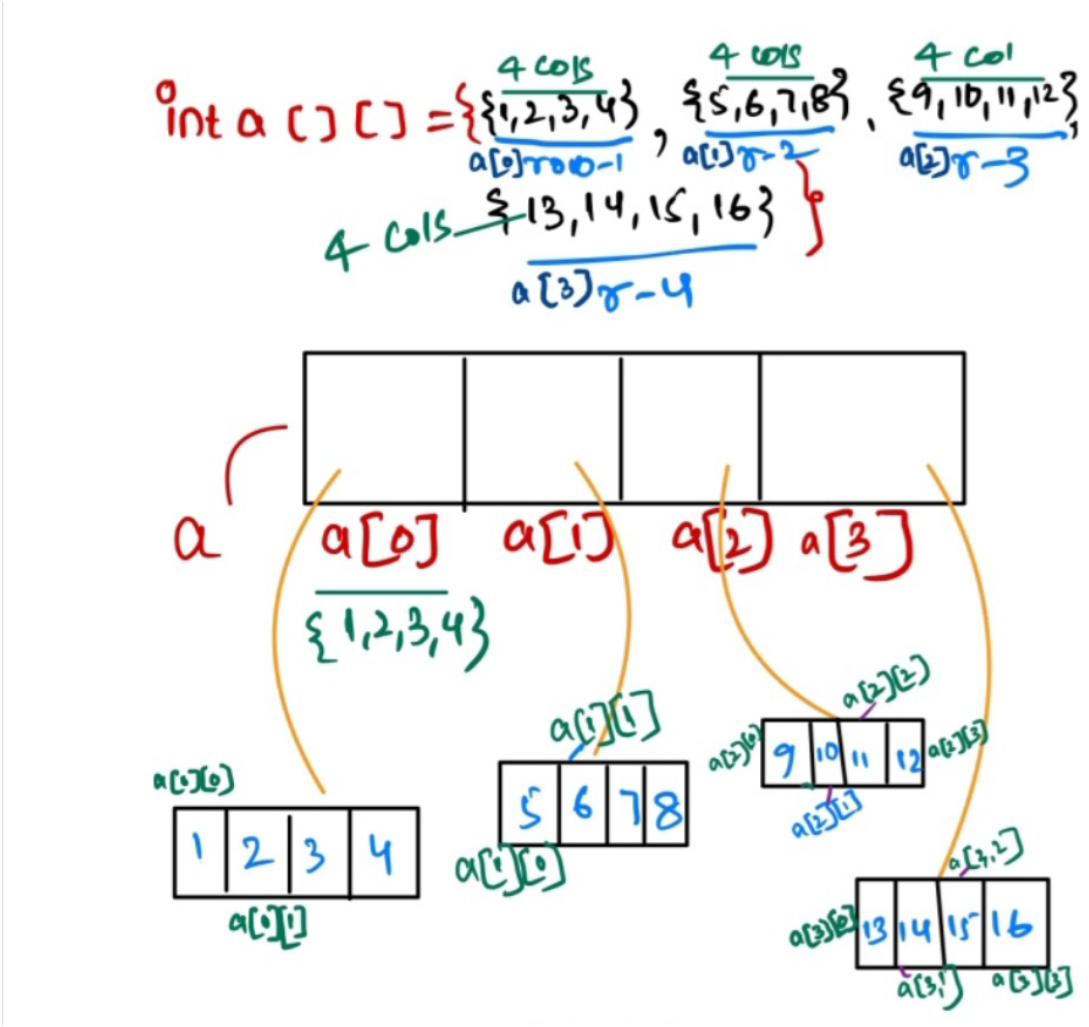
- `int` - datatype
- `arr` - array name
- `[2]` -row.
- `[3]` -column

2.

```
int  row=4; int column=4;
```

```
int arr[row][column];
```

Memory Allocation



How to initialize an 2d array?

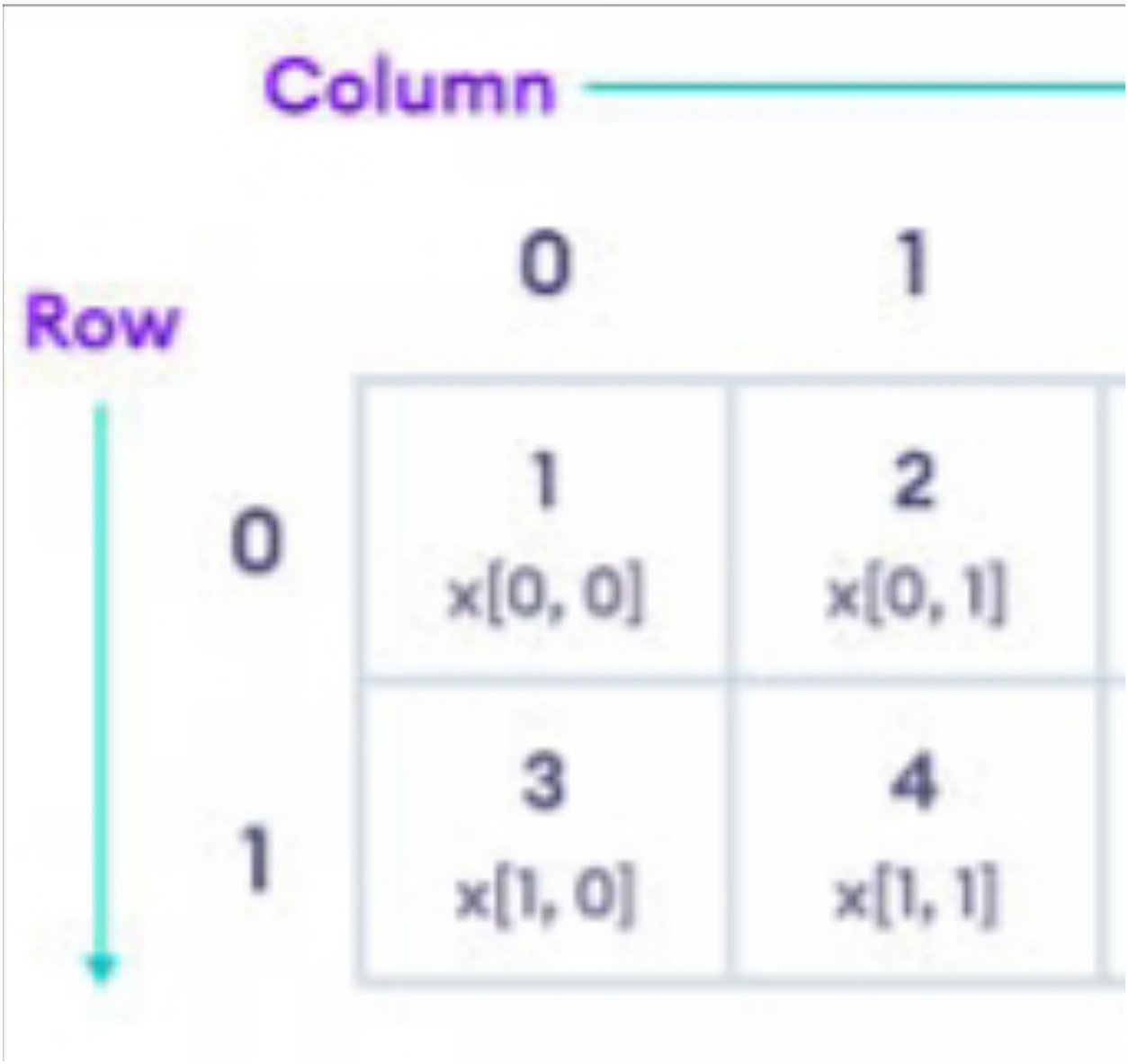
```
int x[2][2];

x[0][0]=1;

x[0][1]=2;

x[1][0]=3;

x[1][1]=4;
```



Declare and initialize together.

```
int arr[2][3] = { {1, 4, 2},{3, 6, 8} };
int arr[ ][3]= { {1, 4, 2}, {3, 6, 8},{9,10,5}};
```

How to print 2d Array elements?

```
int arr[2][2]= { {1, 4},{3, 6,} };
printf("%d ",arr[0][0]);
printf("%d ",arr[0][1]);
```

```
printf("%d ",arr[1][0]);
printf("%d ",arr[0][1]);

- **Output**
- 1 4 3 6
```

How to take input from user in 2d Array?

```
int row=3; int column=4;

int arr[row][column];

for (int i = 0; i < row; i++) {

for(int j=0;j<column;j++){

    scanf("%d", &arr[i][j]);

}
}
```

How to print 2d Array elements ?

```
for (int i = 0; i < size; i++) {

for(int j=0;j<column;j++){

printf("%d   ", arr[i][j]);

}
printf("\n");
}
```

2D Array Questions

1.WAP to take row and column from user and take input and print matrix?

```
Input :
row=2, column=3.
5  6   7   8  2   1
Output :
5  6  7
8  2  1
```

2.wap to take input in 2d array and find the sum of all element ?

```
Input :
row=2, column=3.
5  6   7   8  2   1
Output :
Sum= 29
```

3.wap to take input in 2d array and find the multiply of all element ?

```
Input :
row=2, column=2.
3  2  4  1
Output :
Multiple= 24.
```

4.WAP to calculate the sum of two matrices?

```
Input:
1 2         5  6
3 4         7  8
Output :
6   8
10  12
```

5.WAP to calculate the multiply of two matrices?

```
Input:
1 2         5  6
3 4         7  8
Output :
19  22
43   50
```

6.WAP to take a input and count that number present in a matrix?

```
Input:
1 2 3
3 4 5
K=3
Output :
Count=2
```

7.WAP to take a input from user and find at which index that number is present in the matrix?

```
Input:
8 6 1
4 2 5
K=6
Output :
6 present on [0,1].
Input:
8 6 1
4 2 5
K=12
Output :
12 not present.
```

8.WAP take two matrix and check whether they are equal or not.

```
Example 1.
i/p= [0],[0] : 1
[0],[1] : 2
[1],[0] : 3
[1],[1] : 4
second matrix :
[0],[0] : 1
[0],[1] : 2
[1],[0] : 3
[1],[1] : 4
O/p= Both matrix  Equal.
Example 2.
```

```
i/p= [0],[0] : 11
[0],[1] : 21
[1],[0] : 32
[1],[1] : 41
second matrix :
[0],[0] : 31
[0],[1] : 42
[1],[0] : 25
[1],[1] : 65
O/p= Both matrix are not  Equal.
```

9.WAP to print the transpose of a matrix. The transpose of a matrix is found by interchanging its rows into columns or columns into rows.

```
Input:
1 2 3
4 5 6
Output:
1 4
2 5
3 6
```

10. Write a C program to read elements in a matrix and find the sum of elements of each row and columns of matrix. C program to calculate sum of rows and columns of matrix.

```
Input:
1 2 3
4 5 6
7 8 9
Output:
Sum of row 1 = 6
Sum of row 2 = 15
Sum of row 3= 24
Sum of column 1= 12
Sum of column 2= 15
Sum of column 3= 18
```

11.WAP to take a matrix and find maximum element?

Input:

```
12   21   35
76   10    8
Output:
Max=76.
```

12.WAP to take a matrix and find minimum element?

```
Input:
12   21   35
76   10    8
Output:
Max=8.
```

13.WAP to take a matrix and arrange in ascending order?

```
Input:
12   21   35
76   10    8
Output:
8  10  12
21 35  76
```

14.wap to take a matrix and arrange in descending order?

```
Input:
12   21   35
76   10    8
Output:
76  35  21
12  10   8
```

15.WAP to take a matrix and find **k**th maximum number?

```
Input:
12   21   35
76   10    8
K=3
Output:
3rd max=35
```

16.wap to take a matrix and find **k**th minimum number?

Input:

```
12   21   35
76   10    8
K=3
```

Output:

```
3rd min=12
```