

# **ED1021 - Introduction to computation and visualisation**

## **L8 - Arrays in C**

**Ramanathan Muthuganapathy (<https://ed.iitm.ac.in/~raman>)**

**Course web page: <https://ed.iitm.ac.in/~raman/introcomp.html>**

**Moodle page: Available at <https://courses.iitm.ac.in/>**

# When are they used

- Variable can take only a single value at a time.
- The same variable can take another value during the execution.
  - `int a = 5;`
  - `a = 78;`
- Array variable - can take multiple values at the same time.
- Vectors (1D), Matrices (2D), ..... nD (Linear Algebra will use them quite regularly)

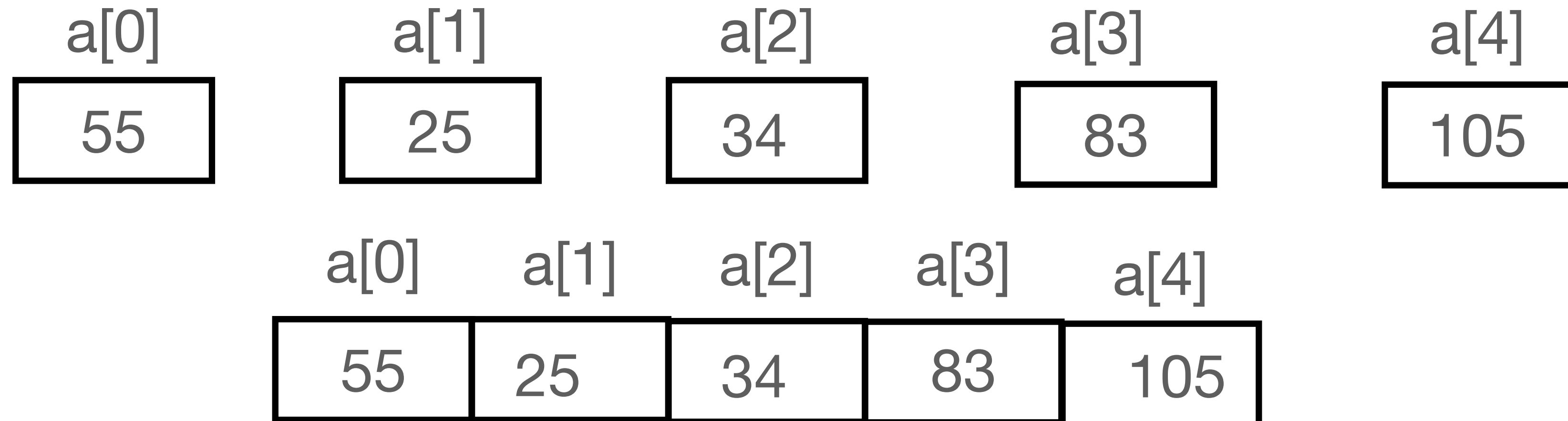
# Declaring an array variable

- `int a[5], b;`
  - `a` - Array variable
  - `[]` - index or array subscript (integer, usually positive, but negative also possible!)
  - '`a`' has five locations where as '`b`' has only one.



# Integer array or array of integer values

- `int a[5];`
  - `a` is an array variable which can take only integer constants
  - starting value for array subscript is 0 and end is one less than its size.



# Floating point array or array of floating point values

## Marks of five subjects.

- `float mark1, mark2, mark3, mark4, mark5;`
- `float marks[5];`
  - 'marks' is an array variable which can take only floating point constants

marks[0]	marks[1]	marks[2]	marks[3]	marks[4]
55.34	25.45	34.68	83.99	105.5

marks[0]	marks[1]	marks[2]	marks[3]	marks[4]
55.34	25.45	34.68	83.99	105.5

# Input for array variable

- `int a[5], b;`
- `scanf("%d", &b);`
- `scanf("%d %d %d %d %d", &a[0], &a[1], &a[2], &a[3], &a[4]);`
- Is there a better way?

# Example program

```
#include <stdio.h>
```

```
int main( )
```

```
{
```

```
    int i, a[10];
```

```
    for (i =0; i < 10; i++) {
```

```
        scanf("%d", &a[i]);
```

```
        printf("%d\n", a[i]);
```

```
    }
```

```
}
```

# Example program

**Adding two row vectors a and b and store them in c**

```
#include <stdio.h>
```

```
int main( )
```

```
{
```

```
    int i;
```

```
    float a[5], b[5], c[5];
```

```
    for (i =0; i < 5; i++) {
```

```
        scanf("%d %d", &a[i], &b[i]);
```

```
        c[i] = a[i] + b[i];
```

```
        printf("%d\n", c[i]);
```

```
    }
```

```
}
```



# Illustration

a[0]	a[1]	a[2]	a[3]	a[4]
55.34	25.45	34.68	83.99	105.5

b[0]	b[1]	b[2]	b[3]	b[4]
5.34	5.45	4.68	3.99	5.5

c[0]	c[1]	c[2]	c[3]	c[4]

**CW: Suppose you have marks for 5 students in ED1021. WAP to take inputs for them and find the average score.**

**HWs (assume x to be an array of floats  
and do this for various values)**

**1)  $1 + x + x^2 + x^3 + \dots$**

**2)  $1 - x + x^2 - x^3 + \dots$**

**3)  $x + x^2 / 2 + x^3 / 3 + \dots$**

**4)  $x - x^2 / 2 + x^3 / 3 - \dots$**

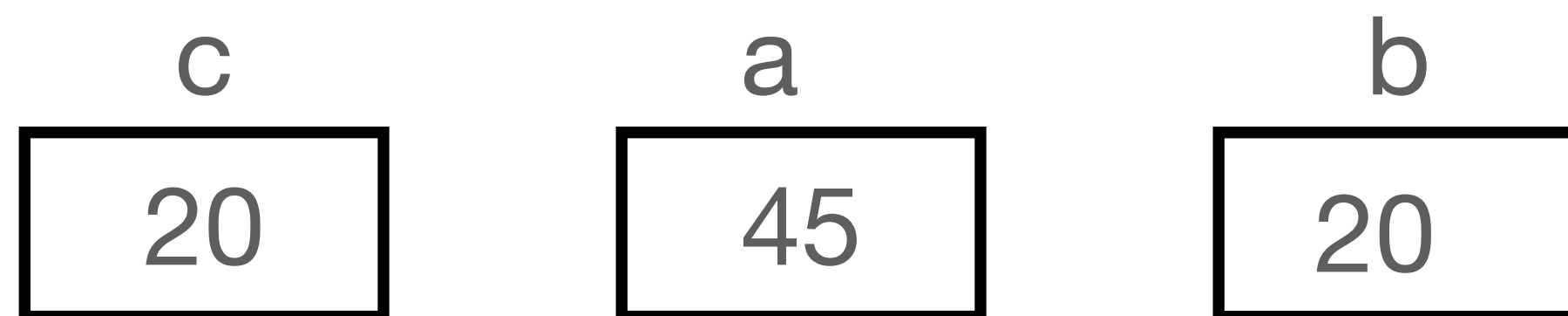
# Some strange behaviours!

Demo using L8\_Arrays\_1D.c

# Swapping two variables

## scalar ones

```
int main() {  
    int a, b, c;  
    a = 20; b = 45;  
    printf("Values of a and b before swapping a = %d, b = %d\n", a, b);  
    c = a;  
    a = b;  
    b = c;  
    printf("Values of a and b after swapping a = %d, b = %d\n", a, b);  
}
```



# Swapping two variables within an array

```
int main() {  
    int c, i;  
    int arr1[6] = {1, 2, 3, 4, 5, 6};  
  
    // Swapping adjacent values in array arr1  
    for (i=0; i<6; i+=2) {  
        c = arr1[i];  
        arr1[i] = arr1[i+1];  
        arr1[i+1] = c;  
    }  
    for (i=0; i<6; i++) {  
        printf("%d\t", arr1[i]);  
    }  
}
```

a[0]	a[1]	a[2]	a[3]	a[4]	a[5]
1	2	3	4	5	6

a[0]	a[1]	a[2]	a[3]	a[4]	a[5]

## HWs (Array)

- 1) Given an array, find its max and min values.
- 2) (Same array) Swap the first and last, second and second-last and so on..
- 3) Identify the odd and even integers present in the array and print their corresponding array indices.

# Higher dimensional arrays

## 2D Array (Matrix)

- `int a[5];` (only one subscript and hence called as 1D Array)
- `int a[3][3];` (Two subscripts and hence 2D array)
- `int a[3][3][3];` (3D array)
- `int a[3][4][5][4][6][7].....;` (nD array)



# 2D Arrays - Matrices

- `int a[5];` Subscript runs actually from 0 to 4
- `int mat[3][2];` row, col

- row from 0 to 2

- col from 0 to 1

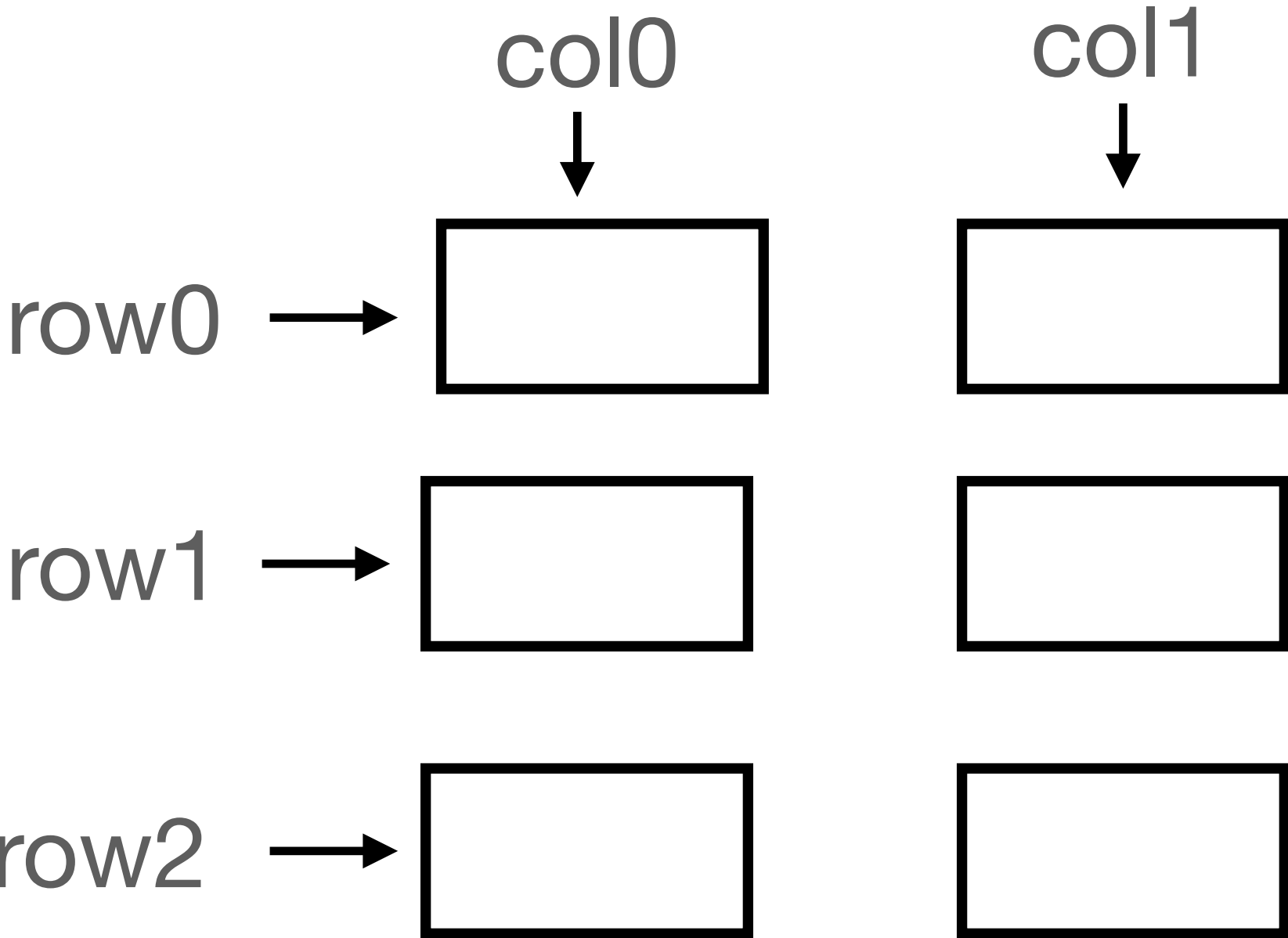
`mat[0][0], mat[0][1],`

`mat[1][0], mat[1][1],`

`mat[2][0], mat[2][1]`

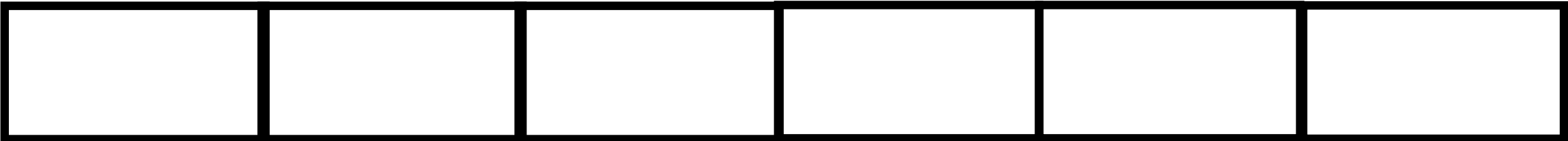
# Pictorial view!

## 2D Array



5	8
10	15
2	2

## In Practise



# 3D Array

**How do you visualise this!**

# Input and output statements

- For 1D array, one loop was used!
- For 2D array?

# Example code

## L8\_Array\_2D.c

```
#include <stdio.h>
int main(){
    float MatA[2][2];
    int i, j;
    //Reading MatA
    for (i = 0; i < 2; i++) {
        for (j = 0; j < 2; j++) {
            printf("Enter input for MatA[%d][%d]: ", i, j);
            scanf("%f", &MatA[i][j]);
        }
    }
    //Writing MatA
    for (i = 0; i < 2; i++) {
        for (j = 0; j < 2; j++) {
            printf("%f\t", MatA[i][j]);
        }
        printf("\n");
    }
    printf("\n");
}
```

# Example code - Matrix addition

## Matrix addition

## HWs (2D Array / Matrices) - WAP

- 1) Take a 2D array
  - (a) find its max and min values.
  - (b) Print only its Diagonal elements.
  - (c) Print off-diagonal elements
- 2) Transpose of a given matrix
- 3) Subtraction of two matrices
- 4) Swap adjacent rows (columns)
- 5) Check a matrix is symmetric or not.

## More difficult ones:

- 1) Matrix Multiplication
- 2) Determinant
- 3) Inverse