Array 2

Basic Programming Teaching Team 2022

Objectives

After studying this material, students should be able to:

- Understand the concept of 2-dimensional arrays
- Provide examples of the use of 2-dimensional arrays
- Complete matrix case studies and others

Definition

- The array we studied earlier is one dimensional, consisting of only one row of elements
- Sometimes data needs to be presented in tabular form, for example a spreadsheet that requires a 2-dimensional array
- Example:
  - Guest book in library which records visitors for a certain time
  - Movie ratings filled in by the audience
    - Each row is rated by a different audience
    - Each column contains a different film

VISITOR SIGN IN School Name

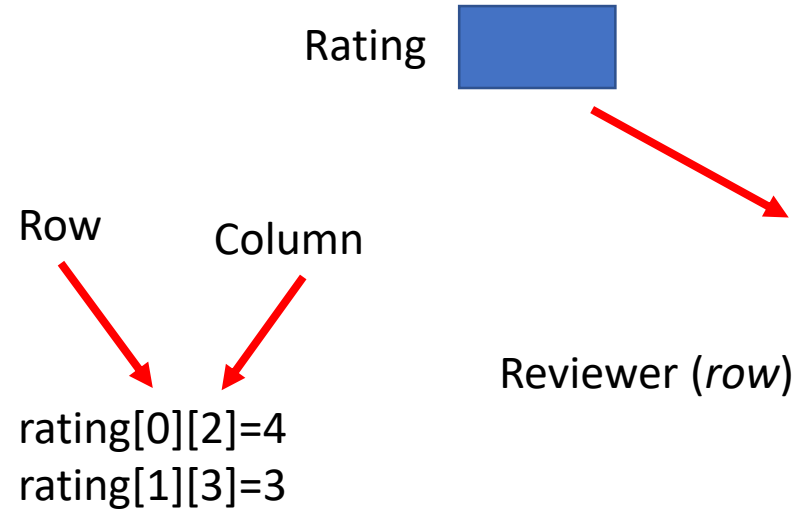
Please check in with the secretary before signing in.

DATE	VISITOR'S NAME	REASON FOR VISIT	TIME IN	TIME OUT	SIGN / INITIAL
5/12	Michael Smith	Volunteer tutoring	4:30	6:00	MJSmith

	Pirates of the Caribbean 4	Kung Fu Panda 2	Harry Potter 6	Harry Potter 7
U_1	4	4	1	2
U_2	3	4	2	1
U_3	2	2	4	4
U_4	4	4	1	?

Definition

- A 2-dimensional array is an array with an index number consisting of 2 numbers, one for rows and one for columns
- Example:



Movie (*column*)

Reviewer (*row*)

	0	1	2	3
0	4	3	4	4
1	1	1	2	3
2	1	2	3	4

2-Dimensional Array Declaration

- To declare a 2-dimensional array variable, the method is the same as a 1-dimensional array. Only differs the number of brackets [ ]
- General form:

```
data_type[][] array_name = new data_type[x][y];
```

x = the number of rows

Y = the number of columns

- Example:

```
int[][] arr = new int[10][20];
```

2-Dimensional Array Declaration

- Another form:

```
1. data_type[][] array_name;
2. data_type [][]array_name;
3. data_type array_name[][];
4. data_type []array_name[];
```

```
int[][] ratings;
int [][]scores;
int students[][];
int []countries[];
```

- You don't need to be confused about declarations, usually the ones that are often used are 1 and 3

2-Dimensional Array Initialization

- 2 Dimensional array initialization is done by assigning value to 2 Dimensional array using assignment operator =
- 2 Dimensional array elements are allowed to be empty
- Example:

```
int[][] a = {
        {1, 2, 3, 4},
        {4, 5, 6, 9},
        {7, 8, 9, 10}
      };
```

```
int[][] a = {
        {1, 2, 3},
        {4, 5, 6, 9},
        {7}
      };
```

	0	1	2	3
0	1 [0,0]	2 [0,1]	3 [0,2]	4 [0,3]
1	4 [1,0]	5 [1,1]	6 [1,2]	9 [1,3]
2	7 [2,0]	8 [2,1]	9 [2,2]	10 [3,3]

	0	1	2	3
0	1 [0,0]	2 [0,1]	3 [0,2]	
1	4 [1,0]	5 [1,1]	6 [1,2]	9 [1,3]
2	7 [2,0]			

```
int[][]ratings=new int[3][4];
ratings[0][0]=1;
ratings[0][1]=2;
ratings[0][2]=3;
ratings[0][3]=4;
ratings[1][0]=1;
ratings[1][1]=2;
ratings[1][2]=3;
ratings[1][3]=4;
```

Enter 2-Dimensional Array Values via Keyboard

```
int[][] ratings = new int[3][4];
Scanner scan = new Scanner(System.in);
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 4; j++) {
        System.out.print("Enter a value for array [" + i + "," + j + "]: ");
        ratings[i][j] = scan.nextInt();
    }
}
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 4; j++) {
        System.out.print(ratings[i][j] + " ");
    }
    System.out.println("");
}
```

```
Enter a value for array [0,0]: 4
Enter a value for array [0,1]: 5
Enter a value for array [0,2]: 5
Enter a value for array [0,3]: 3
Enter a value for array [1,0]: 2
Enter a value for array [1,1]: 1
Enter a value for array [1,2]: 5
Enter a value for array [1,3]: 4
Enter a value for array [2,0]: 5
Enter a value for array [2,1]: 2
Enter a value for array [2,2]: 4
Enter a value for array [2,3]: 5
4 5 5 3
2 1 5 4
5 2 4 5
```

Information :
What if the array size is changed? Of course we also need to change the number of rows and columns in the input loop and when displaying the array.

Print 2-Dimensional Array - for each

```
int[][] ratings = new int[3][4];
Scanner scan = new Scanner(System.in);
for (int i = 0; i < 3; i++) {
    for (int j = 0; j < 4; j++) {
        System.out.print("Enter a value for array [" + i + "," + j + "]: ");
        ratings[i][j] = scan.nextInt();
    }
}
for (int rate[]: ratings) {
    for (int x: rate) {
        System.out.print(x + " ");
    }
    System.out.println("");
}
```

```
Enter a value for array [0,0]: 5
Enter a value for array [0,1]: 3
Enter a value for array [0,2]: 3
Enter a value for array [0,3]: 1
Enter a value for array [1,0]: 4
Enter a value for array [1,1]: 2
Enter a value for array [1,2]: 5
Enter a value for array [1,3]: 2
Enter a value for array [2,0]: 1
Enter a value for array [2,1]: 4
Enter a value for array [2,2]: 3
Enter a value for array [2,3]: 5
5 3 3 1
4 2 5 2
1 4 3 5
```

When displaying all array elements using for-each, it is much simpler to code than before.

2-Dimensional Array Size

- Each array, both 1D and 2D arrays, has a size, using the length attribute.
- Example:

```
int[][] a = new int[3][4];
```

- a.length returns 3, or its row (the first dimension)
- a[0].length returns 4, or its column (second dimension)
- When using attribute / variable **length**, the advantage is that when the array size changes we don't need to change the code to input / display the array.

2 Dimensional Array Size - length

```
int[][] ratings = new int[3][4];
Scanner scan = new Scanner(System.in);
for (int i = 0; i < ratings.length; i++) {
    for (int j = 0; j < ratings[0].length; j++) {
        System.out.print("Enter a value for array [" + i + "," + j + "]: ");
        ratings[i][j] = scan.nextInt();
    }
}
for (int i = 0; i < ratings.length; i++) {
    for (int j = 0; j < ratings[0].length; j++) {
        System.out.print(ratings[i][j] + " ");
    }
    System.out.println("");
}
```

```
Enter a value for array [0,0]: 4
Enter a value for array [0,1]: 5
Enter a value for array [0,2]: 5
Enter a value for array [0,3]: 3
Enter a value for array [1,0]: 2
Enter a value for array [1,1]: 1
Enter a value for array [1,2]: 5
Enter a value for array [1,3]: 4
Enter a value for array [2,0]: 5
Enter a value for array [2,1]: 2
Enter a value for array [2,2]: 4
Enter a value for array [2,3]: 5
4 5 5 3
2 1 5 4
5 2 4 5
```

The example above shows that the code for data input and display data can be dynamic when the array size changes.

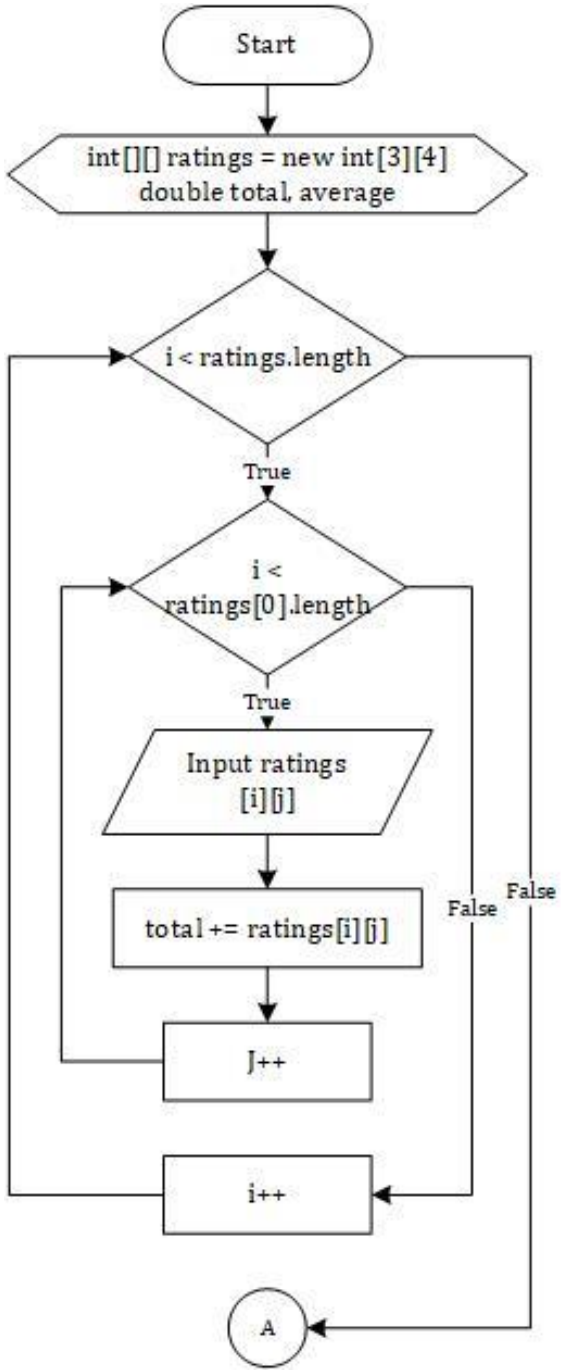
Case Study

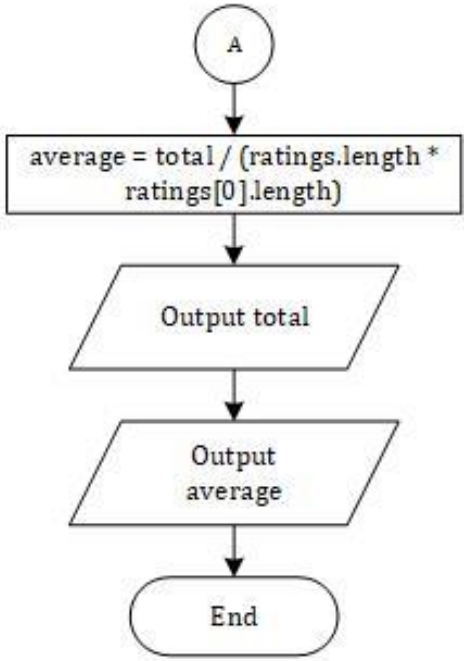
Example 1

- Create a flowchart to calculate the average of the 2-dimensional Array on the film rating table which consists of 3 rows (rating viewers) and 4 columns (film titles)!

Example 1 - Answer

Flowchart





Example 1 - Answer

Program Code

```
int[][] ratings = new int[3][4];
double total = 0;
Scanner scan = new Scanner(System.in);
for (int i = 0; i < ratings.length; i++) {
    for (int j = 0; j < ratings[0].length; j++) {
        System.out.print("Enter a value for array [" + i + "," + j + "]: ");
        ratings[i][j] = scan.nextInt();
        total += ratings[i][j];
    }
}
for (int rate[] : ratings) {
    for (int x : rate) {
        System.out.print(x + " ");
    }
    System.out.println("");
}
double average = total / (ratings.length * ratings[0].length);
System.out.println("Total: " + total);
System.out.println("Average: " + average);
```

```
Enter a value for array [0,0]: 5
Enter a value for array [0,1]: 3
Enter a value for array [0,2]: 3
Enter a value for array [0,3]: 4
Enter a value for array [1,0]: 2
Enter a value for array [1,1]: 1
Enter a value for array [1,2]: 4
Enter a value for array [1,3]: 2
Enter a value for array [2,0]: 5
Enter a value for array [2,1]: 3
Enter a value for array [2,2]: 2
Enter a value for array [2,3]: 3
5 3 3 4
2 1 4 2
5 3 2 3
Total: 37.0
Average: 3.0833333333333335
```

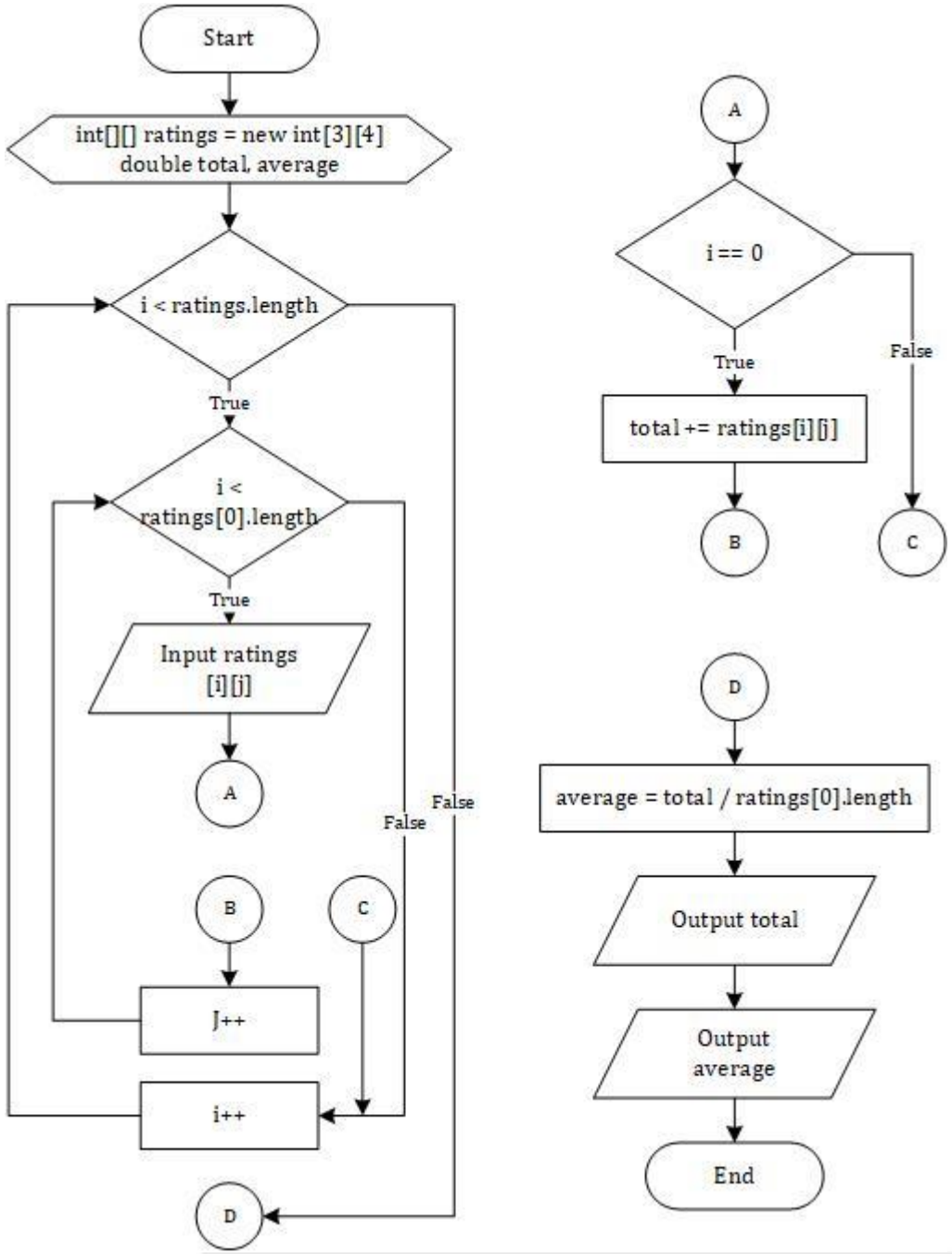
The average value is obtained from the sum of all matrix element values divided by the product of the size of the row and column array

Example 2

- Create a flowchart to calculate the average of certain rows (for example, row 1) in the film rating table which consists of 3 rows (rating viewers) and 4 columns (movie title)!

Example 2 - Answer

Flowchart



Example 2 - Answer

Program Code

```
int[][] ratings = new int[3][4];
double total = 0;
Scanner scan = new Scanner(System.in);
for (int i = 0; i < ratings.length; i++) {
    for (int j = 0; j < ratings[0].length; j++) {
        System.out.print("Enter a value for array [" + i + "," + j + "]: ");
        ratings[i][j] = scan.nextInt();
        if (i==0) {
            total += ratings[i][j];
        }
    }
}
for (int rate[] : ratings) {
    for (int x : rate) {
        System.out.print(x + " ");
    }
    System.out.println("");
}
double average = total / ratings[0].length;
System.out.println("Total: " + total);
System.out.println("Average: " + average);
```

```
Enter a value for array [0,0]: 5
Enter a value for array [0,1]: 3
Enter a value for array [0,2]: 4
Enter a value for array [0,3]: 2
Enter a value for array [1,0]: 5
Enter a value for array [1,1]: 1
Enter a value for array [1,2]: 2
Enter a value for array [1,3]: 4
Enter a value for array [2,0]: 2
Enter a value for array [2,1]: 3
Enter a value for array [2,2]: 4
Enter a value for array [2,3]: 5
5 3 4 2
5 1 2 4
2 3 4 5
Total: 14.0
Average: 3.5
```

Assignment

Adi is a student who every day helps sell cakes made by his mother in the canteen in several campus buildings where Adi studies. Description: Rows showing the canteen in a building; Column shows the number of cakes)

	Pancakes	Pudding	Rainbow Cake	Steamed Buns
Building A	10	25	20	25
Building B	15	23	15	25
Building C	12	12	19	23
Building D	13	10	28	20

The price of a pack of pancakes is IDR 3000, pudding is IDR 2500; rainbow cake is IDR 4000, and steamed buns is IDR 4500. Create a flowchart to calculate:

a. The number of cakes sold in each canteen in buildings A to D
b. The number of each cake in the whole building
c. Total profit if all the cakes in each building are sold out