CST 3613 Java Application Development
Fall 2022

hlocklear@cuny.edu

HW 1

Homework 1

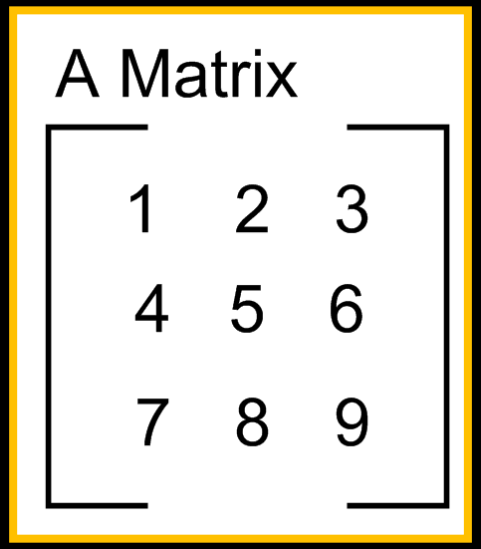
STATIC METHODS AND ARRAYS

Professor HG Locklear

General

- A **Matrix** is a rectangular array of number, symbols, or expressions arranged in rows and columns.
- The individual values in the matrix are known as **entries**.
- Matrices do not have to have the same number of rows and columns.
- We refer to a matrix as **m x n** matrix where:
  - m is the number of rows
  - n is the number of columns
- If a matrix has the same number of rows and columns, we say the matrix is **square**.

- **We can represent a Matrix in Java as a 2D Array of some specified length**.

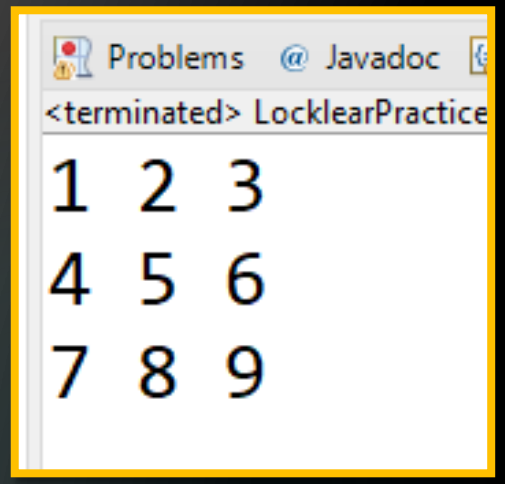


	0	1	2
0	1	2	3
1	4	5	6
2	7	8	9

Create and Display Matrix

TASK #0

Create a static method named **displayMatrix** which accepts a square integer matrix of any length and displays it in rows and columns as shown below.



TASK #1

Create a static method named **buildRandomMatrix** which accepts no parameters and returns a 5 x 5 matrix of randomly-generated integers between 1 and 10 inclusive.

Diagonal Product

TASK #2

Create a static method **diagonalProduct** that accepts a 2D Square Integer Matrix of at least length 5 and returns the product of the sum of the A0 (Left) and A4 (Right) main diagonals (shown in yellow).

Your static method must utilize a for loop to process the matrix and calculate the value.

Your static method must verify that the length of the array passed to the method is of at least length 5 and returns 0 if it isn't.

	0	1	2	3	4
0	1	2	3	4	5
1	6	7	8	9	1
2	2	3	4	5	6
3	7	8	9	1	2
4	3	4	5	6	7

A

(1 + 7 + 4 + 1 +7) x (5 + 9 + 4 + 8 + 3) = 20 x 29 = 580

Sum Columns

TASK #3

Create a static method **sumColumns** that accepts a 2D Square Integer Matrix of at least length 5 and returns a 1D Array containing the sum of columns of the Matrix.

Your static method must utilize one or more **while loops** to process the matrix and create the new 1D array.

Your static method must verify that the length of the array passed to the method is of at least length 5 and returns a 1D array of length 5 filled with 0's if it isn't.

	0	1	2	3	4
0	1	2	3	4	5
1	6	7	8	9	1
2	2	3	4	5	6
3	7	8	9	1	2
4	3	4	5	6	7

A

	0	1	2	3	4
0	19	24	29	25	21

B

Store Even Values

TASK #4

Create a static method **evenOnly** that accepts a 2D Square Integer Matrix of at least length 5 and returns an 1D Array containing only the even number in the Matrix.

Your static method must utilize one or more **for loops** to process the matrix and create the new 1D array.

Your static method must verify that the length of the array passed to the method is of at least length 5 and returns a 1D array of length 5 filled with 0's if it isn't.

	0	1	2	3	4
0	1	2	3	4	5
1	16	7	18	9	1
2	22	3	24	5	26
3	7	28	9	1	32
4	3	34	5	36	7

A

	0	1	2	3	4	5	6	7	8	9	10
0	2	4	16	18	22	24	26	28	32	34	36

B

Binary Conversion

TASK #5

Create a static method **convertToBinary** that accepts **any** 2D Square Integer Matrix containing only 1's and 0's and returns a 1D Array containing the binary conversion value of each row.

Your static method must utilize one or more **for loops** to process the array and create the new array.

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

A

	0	1	2	3	4
0	13	16	27	7	31

B

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

Binary Conversion
31