Cairo University
Faculty of Engineering
Computer Engineering Department
CMP 306 & CMP N306

Spring 2022

Advanced Programming Techniques
Introduction to Java
Exercise Version II

You are required to implement the below set of classes as a part of a system that uses matrices in its operations. Also, you have to write a test main function to show how to use these classes. The classes are described below.

Interface: Addable
This is an interface for any object that can be added to another instance of its type.

Data Members:
No data members

Methods:
1- Add: takes the second term that will be added to the current object and returns the result.

Class: Matrix
This class implements Addable interface

Data Members:
1- Numbers: A 2D array that carries the numbers of the matrix
2- M: number of rows of the matrix
3- N: number of columns of the matrix

Methods:
2- Constructor: takes the number of rows and columns to initialize the data members and allocates a memory for the numbers 2D array accordingly.

3- SetNumbers: takes a 1D array of numbers containing all the values of the matrix recorded row by row. It returns true if setting the matrix numbers has finished correctly, else it returns false.
For example: if the matrix is
1 2
3 4
The input array will be {1,2,3,4}

4- Print: it prints the elements of the matrix each row in a separate line and each row elements are separated by spaces.

5- Transpose: it formulates the transpose $Numbers^T$ of the 2D array numbers by writing the rows of Numbers as the columns of $Numbers^T$. After executing this function, Numbers array should carry its transpose and the number of rows and columns are swapped.
For Example,

The original matrix	The transposed matrix
1 2	1 3 5
3 4	2 4 6
5 6	

Class: IdentityMatrix

This class inherits from class Matrix. It represents the indentity/unit matrix that is a square matrix carrying ones on its diagonal and the other elements are zeros.

Data Members:

No extra data members

Methods:

1- Constructor: takes the number of rows and columns to pass them to its parent class.

2- SetNumbers: overrides the SetNumbers method in Matrix class. It sets the numbers as done in class Matrix and performs extra checks on the special properties of the indentity matrix.

3- Transpose: overrides the Transpose method in Matrix class. As known, the transpose of the identity matrix is the original matrix itself because it's a symmetric matrix.