

**KADIR HAS UNIVERSITY**  
**CE 343 Object Oriented Programming Languages**  
**2016-2017 Fall**

**Homework I – 5 points**

**Due Date: Sunday 27/11/2016 23:59**

Submit your java **source** files (.java) via BlackBoard before the due date.

Maximum 2 students can work together. The file should contain the name of group members.

If your code **fails** to compile, you will get **zero** point.

Any type of **shared** work with different groups will be considered **cheating**. Thus, do **not** share your work.

**Magic Squares**

One interesting application of two-dimensional arrays is *magic squares*. A magic square is a square matrix in which the sum of every row, every column, and both diagonals is the same. Magic squares have been studied for many years, and there are some particularly famous magic squares. In this exercise you will write code to determine whether a square is magic.

File *Square.java* contains the shell for a class that represents a square matrix. It contains headers for a constructor that gives the size of the square and methods to read values into the square, find the sum of a given row, find the sum of a given column, find the sum of the main (or other) diagonal, and determine whether the square is magic. Note that the read method takes a Scanner object as a parameter.

File *Array.java* contains the shell for a class that contains headers for static methods to return a 2D ragged array, determine whether the array is a square and print the array.

File *ArrayTest.java* contains the shell for a program that, in the first part, creates a ragged array and tells whether the array is a square, and in the second part, reads input for squares from keyboard and tells whether each is a magic square.

Following the comments, fill in the remaining code (don't modify the given code).

Note that the -1 tells the test program to stop reading.

In the following, you can find an output example of the program. You are expected to provide the same output with the given input.

```
*** PART 1 ***
Enter n: 4
*** Ragged Array ***
4      3      2      1
4      3      2
4      3
4
It's not a square!

*** PART 2 ***
Enter square size: 3
Enter 9 values: 4 9 2 3 5 7 8 1 6

*** Square 1 ***
4      9      2
3      5      7
8      1      6
Sum of row 0: 15
Sum of row 1: 15
Sum of row 2: 15
Sum of column 0: 15
Sum of column 1: 15
Sum of column 2: 15
Sum of the main diagonal: 15
Sum of the other diagonal: 15
It's a magic square!

Enter square size: 1
Enter 1 values: 10

*** Square 2 ***
10
Sum of row 0: 10
Sum of column 0: 10
Sum of the main diagonal: 10
Sum of the other diagonal: 10
It's a magic square!

Enter square size: 2
Enter 4 values: 1 2 3 4

*** Square 3 ***
1      2
3      4
Sum of row 0: 3
Sum of row 1: 7
Sum of column 0: 4
Sum of column 1: 6
Sum of the main diagonal: 5
Sum of the other diagonal: 5
It's not a magic square!

Enter square size: -1

Game over!
```