

CS 501B – Introduction to JAVA Programming  
Fall 2023 Semester  
Due: 09/29/2023 Friday at 11:59 PM

**Instructions:**

1. You are not allowed to use any package/library unless stated otherwise.
2. A skeleton Assignment2.java file is attached as a stub of code.
3. You are required to make the necessary changes in the Assignment2.java file.
4. We will be testing your code on hidden test cases.
5. Test your code with your own test cases.
6. Please comment your name and CWID in the first two lines of the Assignment2.java file.
7. Add comments in your code about what you are doing.
8. You are required to zip only Assignment2.java file as  
    FirstName\_LastName\_Assignment#.zip (Ex: Roushan\_Kumar\_Assignment2.zip).
9. This assignment covers topics from week 4
10. Students are not allowed to collaborate with classmates and any other people outside.  
    All work must be done individually. Any work having evidence of showing academic  
    dishonesty violation is subjected to zero for the assignment.

**Penalty:**

1. 10 marks will be deducted for invalid format of file / assignment submission.
2. If you submit after the due date then 10 marks will be deducted for every day after the due day.
3. If you submit an assignment after two weeks from the due date then you will get zero marks.
4. You will receive zero, if code doesn't compile / run.

**Questions:**

Each question carries **25** points and total points is **100**.

1. Complete the method `findMinMax(int[] numbers)` that takes an array of integers as its parameter. The method should find both the minimum and maximum values within the array and return them as a new array. The returned array should have the minimum value at the first index and the maximum value at the second index.

**Example 1:**

Input: `findMinMax(new int[] {32, 45, 21, 56, 78, 34, 25, 67, 89, 12})`

Output: `[12, 89]`

**Example 2:**

Input: `findMinMax(new int[] {10, 5, 14, 3, 7})`

Output: `[3, 14]`

2. Complete the method `reverseInPlace(int[] original)` which takes an array of integers as its parameter. The method should reverse the elements within the same array (in-place) without creating any new arrays for the result and return the reversed array.

**Example 1:**

Input: `reverseInPlace(new int[] {1, 2, 3, 4, 5})`

Output: `[5, 4, 3, 2, 1]`

**Example 2:**

Input: `reverseInPlace(new int[] {6, 7, 8})`

Output: `[8, 7, 6]`

3. Complete the method `sumOfDiagonal(int[][] matrix)` which takes a 3x3 matrix as its parameter. The method should calculate and return the sum of the diagonal elements of the matrix. Use loops to compute the sum.

**Example 1:**

Input: `sumOfDiagonal(new int[][] {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}})`

Output: `15`

**Example 2:**

Input: `sumOfDiagonal(new int[][] {{10, 11, 12}, {13, 14, 15}, {16, 17, 18}})`

Output: `42`

4. Complete the method `transposeMatrix(int[][] matrix)` which takes a 2x3 matrix as its parameter. The method should compute the transpose of the matrix and return it. Use loops to compute.

**Example 1:**

Input: `transposeMatrix(new int[][] {{1, 2, 3}, {4, 5, 6}})`

Output:

```
[
    [1, 4],
    [2, 5],
    [3, 6]
]
```

**Example 2:**

Input: `transposeMatrix(new int[][] {{7, 8, 9}, {10, 11, 12}})`

Output:

```
[
    [7, 10],
    [8, 11],
    [9, 12]
]
```