

**ICS 202 – Data Structures and Algorithms,
Spring Semester 2024/2025 (242)
Assignment #2
Programming Part**

Question I. (40 points) Develop and implement a new class CDLList for a Circular Doubly Linked List (without extending any existing classes). The list should be circular, meaning the last node should point to the first node, and vice versa. The nodes in this list are generic, distinct, and ordered by natural ordering (i.e., according to the `Comparable` interface). Try to implement all methods with a time complexity of $O(n)$ or better. Do not use any existing Java collection classes (e.g., `ArrayList`, `LinkedList`). You may see the `DLL.java` provided in LAB-3. Your class should have the following methods:

- a) `boolean add(T el)` - Adds element `el` in its proper position to maintain the order if it does not already exist. Returns `true` if the element is added, otherwise returns `false`.
- b) `boolean delete(T el)` - Deletes element `el` from the list if it exists. Returns `true` if the element is deleted, otherwise returns `false`.
- c) `T findMiddle()` - Returns the middle element of the list. If the list has an even number of elements, return the first of the two middle elements.
- d) `void reverse()` - Reverses the order of elements in the list.
- e) `int size()` - Returns the number of elements in the list.
- f) `void display()` - Displays all elements in the list

Thoroughly test your implementation with at least 4 different test cases.

Question II (40 points): Use the Queue class implementation provided in LAB-4, you are required to create a method called `transformSentence(String input)`. This method serves as a text processing tool that rearranges words within specifically marked sections of a sentence. These sections are enclosed in curly braces `{` and `}`, and the words inside are to be rotated by a specified number of positions. It uses a Queue to collect words from each section, rotate the words to the left by the given number of positions, and reinsert the rotated words back into the sentence. It should also handle sections with variable-length words and multiple spaces between words. Few test cases are as follows:

Input	Expected Output
Check this {example out 1} now	Check this out example now
Here is {another tricky test 2} example	Here is test another tricky example
{one two three four 3} outside {five six 1}	four one two three outside six five
No curly braces here!	No curly braces here!
Unmatched {example 2 or missing number}	Unmatched {example 2 or missing number}

IMPORTANT NOTE REGARDING THIS HOMEWORK SUBMISSION

This homework will be submitted to GRADESCOPE under **Assignment-2-Programming**. For each question, please do the following:

- 1- Submit all your java classes with your name and ID as the first comment on each source file.
- 2- Submit snapshots/images of running your program on different inputs.
- 3- Submit your code as a zip file.

Assignment #2

Written Part

Question III (10 points): Evaluate the following postfix expression using a stack, showing all the intermediate steps: $9\ 3\ 5\ *\ +\ 4\ +\ 7\ /\ 6\ *\ 9\ -$

Question IV (10 points):

Consider a queue implemented using a circular array with a maximum capacity of 5 elements. Initially, the queue is empty.

- Perform the following sequence of operations and show the state of the queue after each operation: Enqueue(10), Enqueue(20), Enqueue(30), Dequeue(), Enqueue(40), Enqueue(50), Enqueue(60), Dequeue(), Enqueue(70).
- Explain why the last enqueue operation may or may not be successful.
- What are the advantages of using a circular array implementation for a queue compared to a linear array implementation?

IMPORTANT NOTE REGARDING THIS HOMEWORK SUBMISSION

Submit your solutions for the written part under **Assignment-2-Written**. Your homework submission must be a pdf file. If you do not type your homework and just solve by hand, make sure you clearly scan the pages (using a scanner or a mobile scanning software like Office Lens or Cam Scanner) of your homework and convert it to **A SINGLE pdf file** or using the GRADESCOPE app directly on iOS/Android. Make sure to **map each answer to the corresponding question**. Failure to do the mapping will result in losing 2 points.