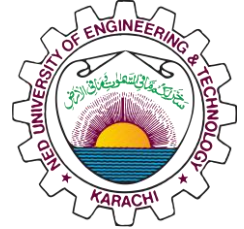




**NED University of Engineering & Technology**  
**Department of Computer Science & Information Technology**  
**CS-159: Data Structures & Algorithms**  
**Fall 2025**



**ASSIGNMENT # 2**

**Submission Deadline: Oct 13<sup>th</sup>, 2025.**

**Submission Guidelines**

Please carefully read the following instructions for submission of the assignment.

- Please submit the assignment before the deadline. It should be clear that submission after due date would not be considered.
- In case plagiarism strict actions will be taken You are advised to avoid submission of copied solution from any other student.
- **Submission:** Submission will only be accepted through GOOGLE CLASSROOM. You need to submit your work in **two** parts. Submission will be incomplete without both the parts.
  1. A single pdf file that contains solutions to all questions. Each answer is supposed to have properly intended and commented code.  
Before submission, rename your pdf file with your roll number.
  2. C/C++ program files for all questions. The file should be renamed as Q1\_CT01 for the for the 1<sup>st</sup> program if roll no. 01 and so on...

**Question 1:**

Given a string containing just the characters '(', ')', '{', '}', '[', and ']', determine if the input string is valid. An input string is valid if open brackets are closed by the same type of brackets in the correct order.

**Question 2:**

Use a stack to reverse a given string.

**Question 3:**

Write a program that takes a decimal number and uses a stack to convert it to its binary representation.

**Question 4:**

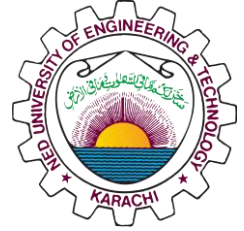
Implement an algorithm that converts an arithmetic expression from **infix notation** to **postfix notation** using a stack. For example,  $(A + B) * C$  becomes  $AB+C*$ .

**Question 5:**

Write a function that evaluates the postfix expression  $P = 5\ 6\ 2\ +\ *\ 12\ 4\ /\ -$



**NED University of Engineering & Technology**  
**Department of Computer Science & Information Technology**  
**CS-159: Data Structures & Algorithms**  
**Fall 2025**



**Question 6:**

Design a stack that, in addition to the standard push and pop operations, also has a `get_min` operation that returns the minimum element. All three operations should have a time complexity of  $O(1)$ .

**Question 7:**

Given an array of integers representing the heights of a histogram's bars, where each bar has a width of 1, find the area of the largest rectangle in the histogram. Use a stack-based approach to solve this in  $O(n)$  time.

**Question 8:**

Implement a queue using a list or an array. Include the basic operations: enqueue (add an element), dequeue (remove an element), peek (view the front element), and `is_empty`.

**Question 9:**

Implement a queue's enqueue and dequeue operations using only two stacks.

**Question 10:**

Given an array and a window size  $k$ , find the maximum element in each sliding window of size  $k$ . Use a deque (double-ended queue) to solve this problem efficiently in  $O(n)$  time.

**Question 11:**

You are given a queue with  $N$  elements. You can perform two operations: Left-Shift (dequeue from the front) and Right-Shift (dequeue from the rear). Implement an algorithm to find the minimum number of operations to make the queue empty.