

Assignment-3

on

ICE 2122: Data Structures and Algorithms Lab

(Stack Queue and Recursion)

Statements

Exp. 01 Dual Mode List Processor (Stack / Queue)

You are to write a program that maintains a dynamic list whose behavior changes based on a selected mode. The program must process the following commands:

- **s** — Switch to **STACK MODE**. All operations behave like a stack (LIFO).
After switching, display the current contents of the stack from **top** → **bottom**, separated by spaces (exclude sentinel).
- **q** — Switch to **QUEUE MODE**. All operations behave like a queue (FIFO).
After switching, display the current contents of the queue from **head** → **tail**, separated by spaces (exclude sentinel).
- **Any integer (e.g., 10, -4, 22)** —
Insert the integer according to the current mode:
 - ✓ In **STACK MODE**: **push** onto the top
 - ✓ In **QUEUE MODE**: **enqueue** to the tailAfter insertion, print the updated structure.
- **p** —
Remove an element according to the current mode:
 - ✓ In **STACK MODE**: **pop** the top
 - ✓ In **QUEUE MODE**: **dequeue** the headPrint the removed value or "EMPTY" if no element can be removed.
- **Q** — Quit the program. All dynamically allocated memory should be properly freed.

Exp. 02 Write a program that stores n numbers in an array, and sorts them by using the quick sort algorithm.-

Application of Stack

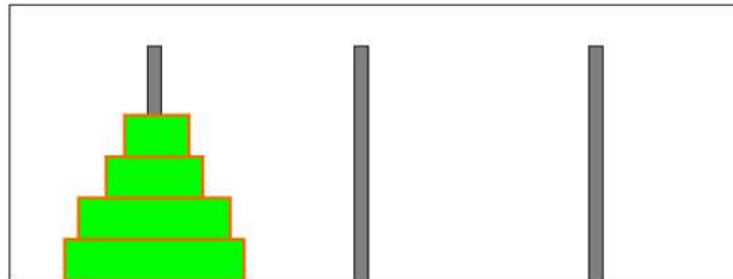
Exp. 03 Write a program to evaluate a given arithmetic expression (infix, postfix, prefix expressions)-

Application of Stack.

Exp. 04 Write a program to transform arithmetic expression:

- Prefix expression to infix expression and vice-versa
- Postfix expression to infix expression and vice-versa
- Prefix expression to postfix expression and vice-versa

Exp. 05* The Tower of Hanoi puzzle was invented by the French mathematician Edouard Lucas in 1883. You are given a tower of 8 disks (the picture below just shows 4 disks for the interest of spaces), initially stacked in decreasing size on one of the three pegs. Write a program to transfer the entire tower to one of the other pegs (the third one in the picture below), moving only one disk at a time, and never a larger one onto a smaller application **of Recursion**



* The problem can be solved recursively.

Please complete Assignments 9th December 2025