

## Assignments (Lab 7)

**NB:** This paragraph is common for all the questions. The programs should work for any value of N (as high it may be). What is the complexity of the algorithms? Please state the reason with a proper explanation [Write on a white paper and submit the scan copy with the assignment PDF file]. Each program should be run for at least TWO test cases. If you are assigning any memory through malloc() function, remember to free() up that memory at the end of the program.

1. Implement a Circular QUEUE using an array.
  
2. Implement a Parking Lot problem using a **Header Linked List based QUEUE** with the following properties.
  - Cars can enter from the REAR end and leave from the FRONT end. Cars are given sequential integer tokens.
  - The user should take input for the Parking Lot size (MAX size) at the beginning.
  - The header node in the header linked list should be used to store the current QUEUE size.
  - While the cars enter to the QUEUE, if the current size exceeds the MAX size, it should give an Overflow error - "Parking Lot FULL".
  - If a car from k-th location wants to leave, all the cars at the FRONT of the k-th car are first removed and put into a STACK. Then, the k-th car is removed. Then, all the cars from the STACK are popped out and again inserted at the FRONT of the queue.
  - At every step, the header node value, FRONT and REAR need to be updated based on the operation.