Stacks & Queues Assignment **Batch: Interview Preparation**

1. Implement basic operations of stack - push, pop & top using array as well as linked list.
2. Implement basic operations of queue – dequeue, enqueue, & front using array as well as linked list.
3. Implement a stack class with O(1) push, pop and getMinimum() functions.
4. Implement a Queue using two stacks.
5. Implement a stack using two queues
6. Check for duplicate parenthesis in an expression e.g. ((a + b) + ((c+d))) has duplicate parenthesis
7. Given an expression check if brackets are balanced e.g. { a + [b+ (c + d)] + (e + f) }
8. A deque is a data structure consisting of a list of items, on which the following operations are possible:
    a. push(x,d): Insert item x on the front end of deque d.
    b. pop(d): Remove the front item from deque d and return it.
    c. inject(x,d): Insert item x on the rear end of deque d.
    d. eject(d): Remove the rear item from deque d and return it. Write routines to support the deque that take constant time per operation
9. The span si of a stock's price on a certain day i is the maximum number of consecutive days (up to the current day) the price of the stock has been less than or equal to its price on day i. Given input array with all stock prices return the spans
10. Given an Infix Expression with brackets and operators, convert it into Postfix and then evaluate it.
11. Sort a queue in place
12. Given an Infix Expression with brackets and operators, convert it into Postfix and then evaluate it.