Data Structures and Algorithms

Lab 05 – Stack and Queue

Exercises/Tasks:

1. Create a Queue using **Array**. Implement all the required methods (enqueue, dequeue, getFront, getSize, isEmpty, isFull, and printAll (print from front-to-rear order)). Besides, show the functionality of the Queue by calling its different methods in the main method.
   Note: Make the queue circular so that insertion should occur if a place is free anywhere in the array.
2. Create a Queue using **Linked List**. Implement all the required methods (enqueue, dequeue, getFront, getSize, isEmpty, and printAll (print from front-to-rear order)). Besides, show the functionality of the Queue by calling its different methods in the main method.
3. Create a Stack using **Array**. Implement all required methods (push, pop, peek, getSize, isEmpty, isFull, printAll (print values in first-to-last order)). Besides, show its functionality by calling its different methods in the main method.
4. Create a Stack using **Linked List**. Besides, show the functionality of the Stack by calling its different methods in the main method.
5. Implement a function that uses a stack to check if a given string is a palindrome. Then, show its working by calling it in the main method.
6. Write a method that uses a stack to check if a given string of parentheses (including parentheses (), braces {}, and brackets []) is balanced or not. Then, show its working by calling it in the main method.
7. Write a program to sort a stack such that the smallest items are on the top. You can use only one additional stack to hold items temporarily.
8. Create a queue using two stacks. Implement all the required methods of Queue and show their working in the main method.
9. Create a method called **getMinimum** that should return the minimum value in a stack in O (1) or constant time.