1. Write a class MyStack using array and add push, pop, displayStack, isEmpty, isFull functions in the class.
   Accept a string from user. use Mystack class to check whether string is palindrome or not.
2. Accept n numbers from user and store it in a array, replace each element with the addition of every other element without using subtraction operator
   Input: { 1, 2, 3, 4, 6 }
   Output: { 16, 14, 13, 12, 10 }
   ex- value 1 will be replaced by 2+3+4+5
   value 4 will be replaced by 1+2+3+5
3. create an array of size n. use it as a circular queue.
   Keep an accepting number from user till user enters number -1. store these numbers in circular queue. if queue is full then remove one number from front end and then add new number. Display how many numbers are removed from the queue till user enters -1.
4. accept numbers from user and store it in a singly linked list in such a manner that data will be always in the sorted order
   example if the linke list contains numbers 3 -→ 4-→ 10
   if user enters 7 then it should get added after 4.
   o/p 3-→4-→7-→10
5. create a circular linked list. Write function to add a number in the linked list. ask user whether to add before head, at the end or at the particular position and then add the number accordingly.
6. create and display a doubly linked list. accept a position from user and display it in reverse order from that position. if the position is > size of linked list the display message wrong position
7. Accept characters from user store it in an array and sort the array using insertion sort. display how many numbers got shifted to right in each iteration
8. Accept characters from user store it in an array and sort the array using bubble sort. display how many numbers got swapped in each iteration
9. Write a program to accept numbers from user and store it in binary tree. display nmbers using inorder traversal. accept a number from user and check whather the number exists in the tree or not.
10. Write a program to accept numbers from user and store it in binary tree. check whether the tree is binary search tree or not. display appropriate message.