Question 1. Implement a class of integer Stack using a Linked List with the following member functions: push, pop (void), peek, empty, and size as explained in the lecture.

Question 2. Implement a class of char Queue using the concept of array with the following member functions: push, pop (void), peek, empty, and full as explained in the lecture.

Question 3. Implement a class of string Deque using the concept of circular array with the following member functions: push_back, pop_back (void), push_front, pop_front (void), peek_front, peek_back, empty, and full as explained in the lecture.

Question 4. Write a quick sort algorithm as a template function in two different forms: (1) Choose pivot randomly, (2) Choose the median as the pivot.

Question 5. In page 27 of Recursion slides, a program for Hanoi Tower is given which is also explained in the Recursion video playlist. As explained in the video, this program has some errors. Fix it.

Question 6. Write a function that takes two sorted linked lists and merges them into a larger sorted linked list efficiently.

Question 7. Write a Boolean function that takes a string and pushes all the characters of the string into a deque and then checks if the string is Palindrome or not.

Question 8. Given a stack with push(), pop(), empty() operations, delete middle (=n/2) of it without using any additional data structure or any loops.
Hint: void deleteMid(stack<char>& st, int n, int curr=0)

Question 9. Given a positive number n, write a function that generates and prints all binary numbers with decimal values from 1 to n.

Examples:

Input: n = 2
Output: 1, 10

Input: n = 5
Output: 1, 10, 11, 100, 101

Algorithm:
1) Create an empty queue of strings
2) Enqueue the first binary number "1" to queue.
3) Now run a loop for generating and printing n binary numbers.
……a) Dequeue and Print the front of queue.
……b) Append "0" at the end of front item and enqueue it.
……c) Append "1" at the end of front item and enqueue it.

Question 10. Write a recursive function named atoiRecursive that takes an array of characters and its size, assuming all the element of the array are numerical characters. The function returns an integer number corresponding to the string.

```
int main(void) {
   char str[] = "112";
   int n = strlen(str);
   cout<< atoiRecursive(str, n))<<endl;  // it should print 112
   return 0; }
```