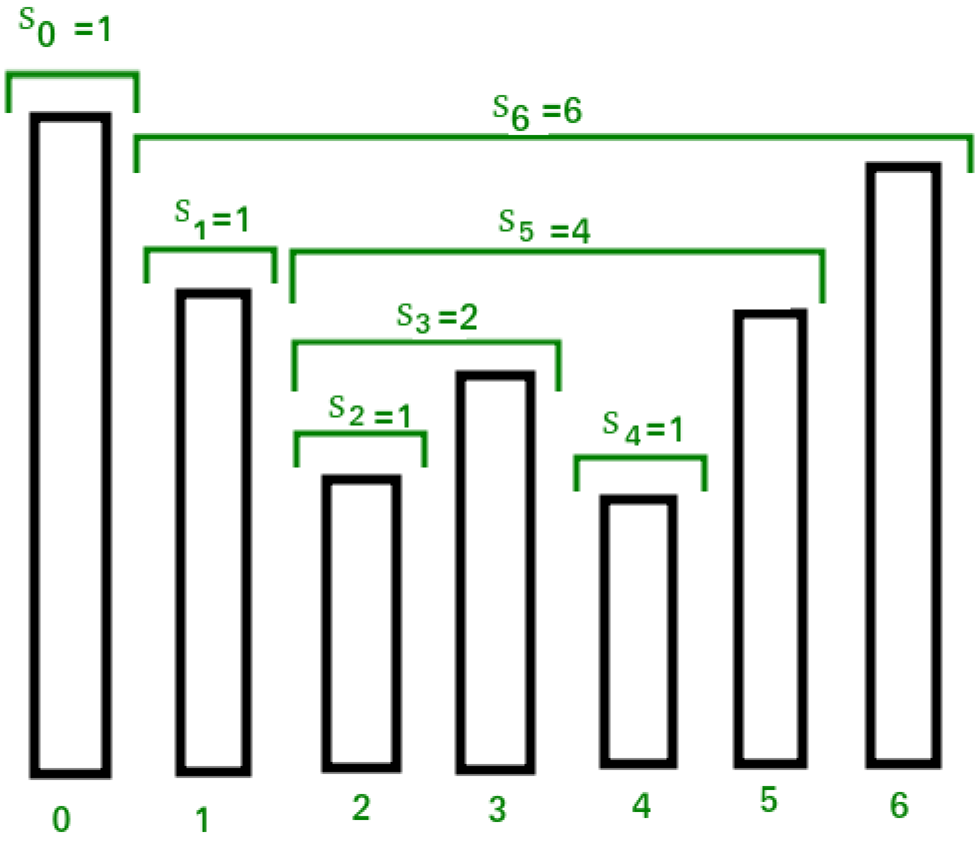
Set – 4

Programming Problems for InfyTQ Round 2

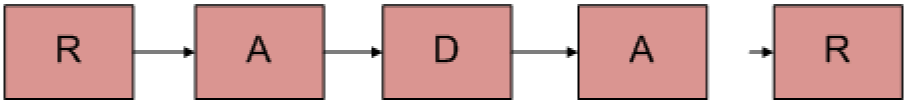
Q1. The stock span problem is a financial problem where we have a series of n daily price quotes for a stock and we need to calculate span of stock's price for all n days.
The span Si of the stock's price on a given day i is defined as the maximum number of consecutive days just before the given day, for which the price of the stock on the current day is less than or equal to its price on the given day.
For example, if an array of 7 days prices is given as {100, 80, 60, 70, 60, 75, 85}, then the span values for corresponding 7 days are {1, 1, 1, 2, 1, 4, 6}



Q2. Given a singly linked list of characters, write a function that returns true if the given list is a palindrome, else false

Example: If following is the input list

Input:



Output: True

Q3. Given two linked lists sorted in increasing order. Merge them such a way that the result list is in decreasing order (reverse order).

Examples:

```
Input:  a: 5->10->15->40

        b: 2->3->20

Output: res: 40->20->15->10->5->3->2

Input:  a: NULL

        b: 2->3->20

Output: res: 20->3->2
```

Q4. Given a Queue consisting of first **n** natural numbers (in random order). The task is to check whether the given Queue elements can be arranged in increasing order in another Queue using a stack. The operation allowed are:

1. Push and pop elements from the stack
2. Pop (Or enqueue) from the given Queue.
3. Push (Or Dequeue) in the another Queue.

Examples :

Input : Queue[] = { 5, 1, 2, 3, 4 }
Output : Yes
Pop the first element of the given Queue i.e 5.
Push 5 into the stack.
Now, pop all the elements of the given Queue and push them to second Queue.
Now, pop element 5 in the stack and push it to the second Queue.

Input : Queue[] = { 5, 1, 2, 6, 3, 4 }
Output : No
Push 5 to stack.
Pop 1, 2 from given Queue and push it to another Queue.
Pop 6 from given Queue and push to stack.
Pop 3, 4 from given Queue and push to second Queue.
Now, from using any of above operation, we cannot push 5 into the second Queue because it is below the 6 in the stack.

Q5. Given an integer k and a queue of integers, we need to reverse the order of the first k elements of the queue, leaving the other elements in the same relative order.
Only following standard operations are allowed on queue.

- enqueue(x) : Add an item x to rear of queue
- dequeue() : Remove an item from front of queue
- size() : Returns number of elements in queue.
- front() : Finds front item.

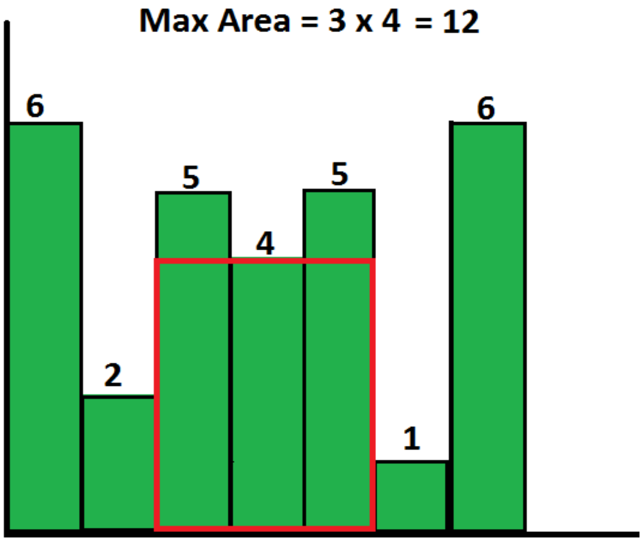
Examples:
Input : Q = [10, 20, 30, 40, 50, 60,
70, 80, 90, 100]
k = 5
Output : Q = [50, 40, 30, 20, 10, 60,
70, 80, 90, 100]

Input : Q = [10, 20, 30, 40, 50, 60,
70, 80, 90, 100]
k = 4
Output : Q = [40, 30, 20, 10, 50, 60,
70, 80, 90, 100]

Q6. Given a singly linked list, write a function to swap elements pairwise. For example, if the linked list is 1->2->3->4->5 then the function should change it to 2->1->4->3->5, and if the linked list is then the function should change it to.

Q7. Find the largest rectangular area possible in a given histogram where the largest rectangle can be made of a number of contiguous bars. For simplicity, assume that all bars have same width and the width is 1 unit.

For example, consider the following histogram with 7 bars of heights {6, 2, 5, 4, 5, 1, 6}. The largest possible rectangle possible is 12 (see the below figure, the max area rectangle is highlighted in red)



Q8. Given pointer to the head node of a linked list, the task is to reverse the linked list. We need to reverse the list by changing the links between nodes.

Examples:
Input: Head of following linked list
1->2->3->4->NULL
Output: Linked list should be changed to,
4->3->2->1->NULL
Input: Head of following linked list
1->2->3->4->5->NULL
Output: Linked list should be changed to,
5->4->3->2->1->NULL
Input: NULL
Output: NULL