

National University of Computer and Emerging Sciences, Lahore Campus



Course:	Data Structures	Course Code:	CS 2001
Program:	BS(CS,SE)	Semester:	Fall 2023
Due Date:	24-09-2023 at 11:59 PM	Total Marks:	40
Type:	Assignment 1	Page(s):	2

Important Instructions:

1. Submit your code in a zip file named as your roll number.
2. You are not allowed to copy solutions from other students. We will check your code for plagiarism using plagiarism checkers. If any sort of cheating is found, negative marks will be given to all students involved.
3. Late submission of your solution is not allowed.
4. All questions / parts must be done keeping in mind the time constraint of $O(n)$ or marks will be deducted.

Question 1:

[20 Marks]

Implement a class (template) SortedSet which is actually a singly linked list with head and tail pointers. It will store data in ascending order while disallowing duplicate data. Implement the following member functions for this class.

[1 mark]

1. Default Constructor. `SortedSet()` ;

[5 marks]

2. An insert function that will insert the data such that resultant set is in ascending order. Duplicate data will not be allowed. `void insert(T const data) ;`

[5 marks]

3. A delete function that will delete the element at the given index. `void delete(int const index) ;`

[3 marks]

4. A print function that will print the contents of the sorted set. `void print() const ;`

[6 marks]

5. A union function that will be passed another sorted set. This function will take union of two sets and store the result in the first set. `void union(SortedSet<T>const &otherSet) ;` For example:
SortedSet a; (suppose it has 1, 2, 3, 4, 10, 50)
SortedSet b; (suppose it has 6, 10, 11)
a.union(b); (a will now contain 1, 2, 3, 4, 6, 10, 11, 50)

Question 2:

[10 Marks]

Using the last(question) linked list, rotate the linked list counter-clockwise by k nodes. Where k is a positive integer and should be taken by the user at the time of the execution.

Hint: You should apply proper checks for the input of k, in case of wrong input by the user display an error message and again input the k until the user enters correct value.

For example:

If the given linked list is 10->20->30->40->50->60 and k is 4, the list should be modified to 50->60->10->20->30->40.

Note: Your code should be generic and it must work for all possible values of k. You can use 'SortedSet' class for the implementation of linked list. Also, you can make a rotate function in 'SortedSet' class for rotating the linked list.

Question 3:

[10 Marks]

You are required to reverse the linked list by changing the links between nodes. Now, implement a reverseList() function which receives a linked list as a parameter but remember this function should not return anything.

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

Note:

1. Don't swap the values, just use pointers to reverse the link list.
2. Also, use 'SortedSet' class from Question no.1 for inserting and printing linked list. Your code should be generic and must fulfill all the required functionalities.



