|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
| C:\Users\saif\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\final design.jpg | **Course:** | **Data Structure** | **Course Code:** | **CS231** |
| **Program:** | **BSCS** | **Semester:** | **3rd** |
| **Name:** |  | **Section:** | **3E& 3F** |
| **Registration #:**  **Due Date:** | **13th Oct, 2020** | **Assessment:**  **Marks:**  **Weightage:** | **Assignment 01**  **20** |
| **Instruction/Notes:** |  | | | |

1. This should be the first page of your submission
2. Plagiarism policy is applicable
3. Use template of code already provided to you in lab session

**Q1: [Singly Linked List] [2+3+3+4=12]**

We have already discussed swapping of first and the last node of a singly linked list in class. For instance the input list and the output lists are shown below

**Input:** 1 -> 2 -> 3 -> 4 -> 5

**Output:** 5 -> 2 -> 3 -> 4 -> 1

a. Write a program in C++ to implement the ***swap*** function that swaps the first and the last node only

b. Modify the swap function to take two inputs, the two arbitrary nodes to be swapped. for instance when called ***swap(3,5)*** , it should swap the third and the fifth node.

**Input:** 11 -> 12 -> 35 -> 41 -> 50

**Output:** 11 -> 12 -> 50 -> 41 -> 35

c. Given a singly linked list, rotate the linked list counter-clockwise by k nodes. Where k is a given positive integer. 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.

d. We can represent two numbers by using two linked lists. Write a function that returns the sum of these two lists. The sum list is linked list representation of the addition of two input numbers. There are two restrictions to solve this problem i.e. you cannot modify the lists and you are not allowed to use explicit extra space *(Resultant list is not explicit)*

**Input:**  
  First List: 1->2->3  // represents number 123  
  Second List: 9->9->9 //  represents number 999  
**Output**:  
  Resultant list: 1->1->2->2  // represents number 1122

**Q2: [Doubly Linked List] [2+3+3=8]**

a. Write a program to concatenate two un-ordered doubly linked lists into one. The input and output of the program is shown as follows

**Input List 1:** 11 <-> 12 <-> 35 <-> 41 <-> 50

**Input List 2:** 44<->21<->56<->5

**Output:** 11 <-> 12 <-> 35 <-> 41 <-> 50<->44<->21<->56<->5

b. Modify the program in (a) to combine two ordered doubly linked lists resulting in single ordered doubly linked lists. The output of the program is shown as follows

**Input List 1:** 11 <-> 12 <-> 35 <->41 <-> 50

**Input List 2:** 44<->21<->56<->5

**Output:** 5<->11 <->12 <-> 21 <-> 35<-> 41<->44<->50<->56

c. Modify the program Q1(d) by using doubly linked list. Write a function that returns the sum of these two lists. The sum list is linked list representation of the addition of two input numbers. There are two restrictions to solve this problem i.e. you cannot modify the lists and you are not allowed to use explicit extra space *(Resultant list is not explicit)* Is it easy to implement? or more difficult?

**Input:**  
  First List: 1<->2<->3  // represents number 123  
  Second List: 9<->9<->9 //  represents number 999  
**Output**:  
  Resultant list: 1<->1<->2<->2  // represents number 1122