National University of Computer and Emerging Sciences, Lahore Campus

	Course:	Data Structure	Course Code:	
	Program:	BSCS	Semester:	4th
	Name:		Section:	4A, 4B
	Registration #:		Assessment	Homework1

Instruction/ Notes: All the images used in this HOMEWORK are uploaded at "SLATE"

The purpose of this homework is to practice basic concepts related to Linked List that we have already covered in class

Q1:

a. Add Two Numbers

You are given two non-empty linked lists representing two non-negative integers. The digits are stored in reverse order and each of their nodes contain a single digit. Add the two numbers and return it as a linked list. [You may assume the two numbers do not contain any leading zero, except the number 0 itself. Moreover, keep in mind the concept of carry in addition]

Example:

Enter the First number: 342

Enter the Second Number: 465

Linked List for the first number: **(2 -> 4 -> 3)**

Linked List for the second number: **(5 -> 6 -> 4)**

Add the Linked Lists: **(2 -> 4 -> 3) + (5 -> 6 -> 4)**

Output: **7 -> 0 -> 8**

Answer: 807

Explanation: 342 + 465 = 807.

Q2: Remove Nth Node From End of List : Given a linked list, remove the n-th node from the end of list and return its head.
Example:

Given linked list: 1->2->3->4->5, and n = 2.

After removing the second node from the end, the linked list becomes 1->2->3->5.

Q3: Merge two sorted lists

Linked List 1 **: 23-> 45-> 65**
Linked List 2 **: 12->67->78**

Merge Linked List: **12 -> 23 ->45 ->65 ->67 ->78**

Q4: Given a linked list, swap every two adjacent nodes and return its head. Example:

Given 1->2->3->4, you should return the list as 2->1->4->3.

Q5: Given a linked list, rotate the list to the right by *k* places, where *k* is non-negative.
Example 1:

Input: 1->2->3->4->5->NULL, k = 2
Output: 4->5->1->2->3->NULL

Explanation:
rotate 1 steps to the right: 5->1->2->3->4->NULL
rotate 2 steps to the right: 4->5->1->2->3->NULL

Q6: Given a sorted linked list, delete all nodes that have duplicate numbers, leaving only distinct numbers from the original list.

Example 1:

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

Output: 1->2->5