NORTH SOUTH UNIVERSITY
Department of Electrical and Computer Engineering
B.Sc. in Computer Science and Engineering Program
Mid Term Assessment
Fall 2021 Semester

Course:	**CSE 225 Data Structure and Algorithms, Section-10**
Instructor:	**Mohammad Rezwanul Huq (MRH1), PhD, Associate Professor (Part-time)**
Full Marks:	**100**
Duration:	**1 hour and 30 minutes [Last time of Submission: 11-20 AM]**

Note: There are **5** (**FIVE**) questions, answer ALL of them. Mark of each question are mentioned at the right margin.

---

1. **Answer** the following questions with brief explanation. Without explanation, your answer will not be accepted. [20]

   (a) What is the runtime complexity of pushFront(int key) in a doubly linked list of integers with a tail but no head?

   (b) What is the runtime complexity of inserting a node before the last node in a singly linked list of integers with a head and a tail pointer?

   (c) What is the runtime complexity of inserting a node before the last node in a doubly linked list of integers with a head and a tail pointer?

   (d) What is the runtime complexity of deleting a node before the last node in a doubly linked list of integers with a head and a tail pointer?

   (e) What is the runtime complexity of inserting a node after a given node (assume that you know the address of the given node) in a singly linked list of integers with a head but no tail?

2. Given a Singly Linked List of integers, **write** a function **removeNodes()** that removes a node if it is at least twice as big as the previous node. The function reads each node only once. While doing this operation, you must not use any additional list or arrays. You may use additional pointers if needed. [20]

   The function prototype is given below.
   ```
   Node* removeNodes (Node* head)
   ```

   For example, the given list is:
   Head → 2 → 5 → 3 → 6 → NULL

   The final list should be: head → 2 → 3 → NULL

   Explanation: 5 is more than twice of 2. Therefore, 5 is removed. 6 is exactly two times of 3. Therefore, 3 is removed.

   Moreover, take an input list of your own containing at least six elements and show the resulting list after invoking the operation. While preparing your input list, make sure that at least two elements and not more than 3 elements will be removed from the list. If your

input and output sequence matched with another student, marks will be deducted automatically.

While writing the function, consider the following Node class.

```
class Node{
    public:
        int key;
        Node* next;
}
```

3. Suppose that, you are given a doubly linked list containing integer values sorted in [20] ascending order. The first node of the doubly linked list is referenced by the HEAD pointer but there is no TAIL pointer. **Write** a function `removeKthNode()` that removes k[th] node from the end of the doubly linked list.

The function prototype is given below.
```
Node* removeKthNode (Node* head, int k)
```

A testcase is given below:
<u>Input List:</u>
```
head --> 1 <--> 2 <--> 3 <--> 4 <--> 5 <--> 6 --> NULL
k = 3
```
<u>Final Output:</u>
```
head --> 1 <--> 2 <--> 3 <--> 5 <--> 6 --> NULL
```

Moreover, **mention** the time complexity of this function in terms of Big-Oh notation and justify your answer.

While writing the function, consider the following Node class.
```
class Node{
    public:
        int key;
        Node* next;
        Node* prev;
}
```

4. (a) Given a Singly Linked List of integers, you want to reverse the list and the reversed list should [20] be stored in another data container such as Stack or Queue. **Which one is more appropriate in this case?** Justify your answer and write a pseudocode to store the reversed list in your preferred data container.

(b) You are working on a problem where you need to frequently access elements at a random location. **Which one should you use – Array or Linked List?** Justify your answer with the runtime complexity expressed in terms of Big-Oh notation.

(c) Consider the following mathematical expression. **Draw** the expression tree.

$$\left(D - \left(C * (A + B)\right)\right) / E$$

5. **(a) Apply** the algorithmic method to change the following expression in postfix notation [20] using a Stack. Show each step of the conversion including stack contents and postfix expression.

$$(A * B) - F * (G / H) / E + C * D$$

(b) Suppose that A = 10, B = The value representing the first two digits of your Student ID, C = Seventh Digit of your Student ID, D = 6, E = 4, F = 5, G = 16 and H = 2.

Determine the value of the expression given in Question 5.(a) using the algorithmic approach facilitating a stack. Show each step of your computation.