University of Central Punjab

Faculty of Information Technology

Mid-Term Exam Spring - 2021

Data Structures and Algorithms – Lab

Instructions for Invigilators:

1. Students will have total **150** minutes to finish the whole exam. It is up to the students to manage their time.

Instructions for Students:

1. Please create file with appropriate name
2. Submit only **.h** and **.cpp** files on portal.
3. Late submissions will **NOT** be considered
4. Create as many classes and functions as required. Remember one function for one functionality.
5. Take care, plagiarism will not be tolerated at any case.
6. No **.Rar/Zip** files are accepted .
7. The paper is close book and close notes. No cheat sheet allowed.
8. Use meaningful variable names, take care of naming conventions and indentation. **5 Marks will be deducted for each thing if not followed.**

Question 1 – 30 Marks

Implement the Linked List using **head** pointer only (you are not allowed to use **tail pointer**). **Interface** (abstract class) of **LinkedList** class is given below. Your task is to provide the complete **implementation** for this question (a child class having name **myLL** is required, this **myLL** class will provide the complete implementation of the LinkedList class)

Interface:

```
template<class T>
class LinkedList
{
	protected:
		Node<T>* head;

	public:
		LinkedList();
		virtual void insertAtEnd(T) = 0;
		virtual T deleteFromHead() = 0;
		virtual bool isEmpty() = 0;
```

```
        virtual void display() = 0;
};
```

Question 2 – 30 Marks

Implement Queue (FIFO) using Linked List implemented in task 1**.**

Interface (abstract class) of **Queue** class is given below (a child class having name **myQueue** is required, this **myQueue** class will provide the complete implementation of the Queue class)

Interface:

```
template<class T>
class Queue
{
protected:
	myLL<T> obj;
public:
	virtual bool isEmpty() = 0;
	virtual void enqueue(T) = 0;
	virtual T dequeue() = 0;
	virtual void display() = 0;
};
```

Question 3 – 30 Marks

Now write a global (non-member) function **reverseQueue** which should reverse all the contents of the **Queue**.

```
template<class T>              //add this line before the function to make it
work as template
Queue <T> reverseQueue(Queue <T> obj);
```

University of Central Punjab

Faculty of Information Technology

Remember: You are not allowed to use any data structure other than the one made in Question 2.

Hint: You can use more than one Queues

Question 4 – 10 Marks

Now test the main function and produce the exact output given below. **It is mandatory to attach the screen shot of your output in your submission (it carries marks).**

```
int main()
{
	cout << "\n\n---------- Best of Luck for the Exam ---------\n\n";
	myQueue<char> q1;
	q1.enqueue('D');
	q1.enqueue('S');
	q1.enqueue('A');
	q1.enqueue(' ');
	q1.enqueue('L');
	q1.enqueue('A');
	q1.enqueue('B');
	q1.display();

	myQueue<char> reverseQ1 = reverseQueue(q1);
	reverseQ1.display();

	return 0;
}
```

Output:

University of Central Punjab

Faculty of Information Technology

Microsoft Visual Studio Debug Console

```
---------- Best of Luck for the Exam ---------

D S A   L A B
B A L   A S D
```