OBJECT ORIENTED PROGRAMMING

LAB TASK 5

16th February 2023

Problem1: Reverse Linkedlist iteratively

1. Create three pointers, prev, current, and next. Set prev and next to null, and current to the head of the linked list.
2. While current is not null:

    a. Set next to the next node of current.

    b. Set the next node of current to prev.

    c. Set prev to current.

    d. Set current to next.

3. Set the new head of the linked list to prev.

Problem2: Removing duplicates from sorted list

1. Set currentNode to the head of the linked list.

2. While currentNode is not null:

    a. Set nextNode to currentNode.

    b. While nextNode's next is not null:

        i. If nextNode's next has the same value as currentNode, set nextNode's next to its next node.

        ii. Otherwise, set nextNode to its next node.

    c. Set currentNode to its next node.

3. Return the modified linked list.

Problem3: Follow following algorithm to implement

Stack using Linked List

1. Define a class for the node of the linked list, which has a data field and a next field that points to the next node in the list.

2. Define a class for the stack, which has a top field that points to the top node of the stack.

3. Implement the push operation:

    a. Create a new node with the given data.

    b. If the stack is empty (top is null), set the new node as the top of the stack.

    c. If the stack is not empty, set the new node's next field to the current top node.

    d. Set the stack's top field to the new node.

4. Implement the pop operation:

    a. If the stack is empty (the top field is null), return null.

    b. Store the data of the top node in a variable.

    c. Set the stack's top field to the next node in the list (if any).

    d. Return the stored data.