Data Structure Lab, B. Tech 2nd Semester

Instructions

1. After completion, you can share the files through google form link.

Assignment -3

1. Write a program in C to insert a new node at the beginning and delete a node from the middle of a Singly Linked List.

Output format:

- *Take user input for linked list*
- *Display the original linked list (Eg. 1 -> 2 -> 3 -> 2 -> 1)*
- *Print the node you want to insert.*
- *Print the linked list after Insertion. (Eg. X -> 1 -> 2 -> 3 -> 2 -> 1)*
- *Print the middle node you want to delete.*
- *Print the linked list after Deletion. (Eg. X -> 1 -> 3 -> 2 -> 1)*

2. Write a C program to Check if the linked list is palindrome or not.

Modify the function to reverse the list and use in you code.

```
void reverse_list(){
    while (curr != NULL){
        // store the next node
      next = curr -> next;
       // reverse the pointer of the current node
       curr ->next = prev;
       // move prev pointer to the current node
       prev = curr;
       // move current to its next node
       curr = next;
}
```

Output format:

- *Take user input for linked list*
- *Display linked list in linked list format. (Eg. 1 -> 2 -> 3 -> 2 -> 1)*
- *Print If palindrome or not.*

3. Create a linked lists and find the intersection of both the linked lists.

Output format:

- *Take user input for Linked list1*
- *Display the original linked list 1 (Eg. 1 -> 2 -> 3 -> 23-> 1)*
- *Take user input for linked list2*
- *Display 2nd linked list (Eg. 11 -> 2 -> 3 -> 23-> 10-> 7)*
- *Display the linked list after intersection. (Eg. 2 -> 3 -> 23)*

NOTE: While creating Linked List, all node values should be taken from user input.

Note: It is requested that everyone should already download a copy of the previous lab programme on their laptops so that we avoid wasting time downloading the programme during the evaluation.