Practical no: 4

Objective: Write a program in C to reverse singly linked list

Program Codes: Following is the code of this problems in C:-

1. Practical5.c

```
#include<stdio.h>
#include<stdlib.h>

typedef struct node node;

struct node{
    int data;
    node* next;
};

void reverse(node**);
void append(node **);
void display(node*);

// prints the list
void display(node* temp){
    printf("\nlist: ");
    while(temp→next≠NULL){
        printf("%d--->", temp→data);
        temp=temp→next;
    }
    printf("%d\n",temp→data);
}

// creates new nodes OR adds in the last
void append(node **adress_of_head){
    node* new_node_ptr = (node*)malloc(sizeof(node));
    new_node_ptr→next = NULL;
    printf("\nEnter data for the node:");
    scanf("%d", &new_node_ptr→data);
```

```
    if(* adress_of_head == NULL){
        * adress_of_head = new_node_ptr;
        return;
    }
    node* temp = *adress_of_head;
    while(temp→next≠NULL){
        temp=temp→next;
    }
    temp→next = new_node_ptr;
}

void reverse(node **adress_of_head){
    // `last` pointer points node just before temp
    // `second_last` pointer points node just before `last` pointer
    // `temp` is used for traversing
    node* last, *second_last, *temp;
    last = second_last = NULL;
    temp = *adress_of_head;
    while(temp≠NULL){
        // sliding pointers one by one
        second_last = last;
        last = temp;
        temp = temp→next;

        // reverting direction of links using `last` and `second_last` pointer
        last→next = second_last;
    }
    // set head after changing link direction
    *adress_of_head = last;
}

void main(){
    node* head = NULL; int choice;
    printf("\nHow many nodes you want?:");
    scanf("%d",&choice);
    for (int i = 0; i < choice; i++)
        append(&head);

    printf("\nBefore reversing ");display(head);
    reverse(&head);
    printf("\nAfter reversing ");display(head);
}
```

Output: Following is the output of the program:-

1.
```
C:\Users\DV yadav\Desktop>gcc Practical5.c && a

How many nodes you want?:5

Enter data for the node:1

Enter data for the node:2

Enter data for the node:3

Enter data for the node:4

Enter data for the node:5

Before reversing
list: 1--->2--->3--->4--->5

After reversing
list: 5--->4--->3--->2--->1
```