**Assignment-3**

**WAP a program in c to calculate the sum of k numbers of node in linked list .**

**Explanation:**

The program calculates the sum of the first k nodes in a singly linked list. Here's the step-by-step working procedure:

1. The program starts by creating a singly linked list with nodes containing integers 10 to 70 using the insertNode function.

2. The insertAtBegin function is used to insert a new node with data 1 at the beginning of the list.

3. The display function is called to print the linked list.

4. The sum function is called with the head of the list and k=4 as arguments.

5. Inside the sum function:

- A temporary pointer temp is set to the head of the list.

- An integer sum is initialized to 0.

- A while loop runs k times (in this case, 4 times).

- In each iteration, the data of the current node is added to sum, and temp is moved to the next node.

- After the loop, the sum of the first k nodes is printed.

The working procedure of the sum function can be visualized as follows:

- temp points to the head of the list (node with data 1).

- In the first iteration, sum becomes 1, and temp moves to the next node (node with data 10).

- In the second iteration, sum becomes 11 (1 + 10), and temp moves to the next node (node with data 20).

- In the third iteration, sum becomes 31 (11 + 20), and temp moves to the next node (node with data 30).

- In the fourth iteration, sum becomes 61 (31 + 30), and temp moves to the next node (node with data 40).

- After the loop, the sum of the first 4 nodes (61) is printed.

// ..................Singly linked list............

#include<stdio.h>

#include<stdlib.h>

//................Struct node...................

typedef struct Node

{

    int data;

    struct Node \* next;

}sll;

//....................Prototype...................

sll \* createNode(int);

sll \* insertNode(sll\*,int);

sll \* insertAtBegin(sll\*,int);

sll \* sum(sll\*,int );

void display(sll \*);

// ..................Main Function................

int main(){

    sll \* head= NULL;

    head = insertNode(head,10);

    insertNode(head,20);

    insertNode(head,30);

    insertNode(head,40);

    insertNode(head,50);

    insertNode(head,60);

    insertNode(head,70);

    head= insertAtBegin(head,1);

    display(head);

    sum(head,4);

    return 0;

}

// ...........Creation of node........................

sll \* createNode(int data){

    sll \* newNode= (sll \*)malloc(sizeof(sll ));

    newNode->data=data;

    newNode->next=NULL;

    return newNode;

}

//...............Insert at end.........................

sll \* insertNode(sll \* head,int data){

    if (head==NULL){

        head= createNode(data);

    }else{

        sll \* temp=head;

        while(temp->next){

            temp=temp->next;

        }

        temp->next=createNode(data);

    }

    return head;

}

//.................Insert at begining....................

sll \* insertAtBegin(sll \*head,int data){

    sll \*temp = head;

    head = createNode(data);

    // head->data =data;

    head->next =temp;

    return head;

}

//...............Display the linked list.................

void display(sll \* head){

    sll \* temp= head;

    while (temp){

        printf("%d ",temp->data);

        temp=temp->next;

    }

}

// ...................Sum of numbers......................

sll \* sum(sll \*head, int k){

    sll \*temp =head;

    int sum =0;

    while(k>0){

        sum += temp->data;

        temp = temp ->next;

        k--;

    }

    printf("\nSum of all element :%d",sum);

}