7. Design, Develop and Implement a menu driven Program in C for the following operations on SLL of Student Data
   with the fields :
   USN, Name, Branch, Sem, PhNo.

   a. Create a SLL of N Students Data by using front insertion.

   b. Display the status of SLL and count the number of nodes in it

   c. Perform Insertion and Deletion at End of SLL

   d. Perform Insertion and Deletion at Front of SLL

   e. Demonstrate how this SLL can be used as STACK and QUEUE

   f. Exit

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

struct node{
        char usn[20],name[10],branch[5];
        unsigned long long int phno;
        int sem;
        struct node *link;
};

typedef struct node *NODE;
NODE temp,FIRST=NULL;

NODE getnode(){
        NODE x;
        x=(NODE)malloc(sizeof(struct node));
        x->link=NULL;
        return x;
}

void read(){
        temp=getnode();
        printf("Enter USN : ");
        scanf("%s",temp->usn);
        printf("Enter the name :\n");
        scanf("%s",temp->name);
        printf("Enter Branch : ");
        scanf("%s",temp->branch);
        printf("Enter the phone number :\n");
        scanf("%llu",&temp->phno);
        printf("Enter the semester : ");
        scanf("%d",&temp->sem);
}

void insert_front(){
        read();
        if(FIRST==NULL)
                FIRST=temp;
        else{
                temp->link=FIRST;
                FIRST=temp;
        }
} void create_SSL(){
        int n,i;
        printf("Enter the number of students : ");
        scanf("%d",&n);
        for(i=1;i<=n;i++){
                printf("Enter the details of the student %d\n",i);
                insert_front();
        }
}

void display_count(){
        int count=1;
        temp=FIRST;
        printf("Student details :\n");
        if(FIRST==NULL)
                printf("Student detail is NULL and the count is 0\n");
        else{
                printf("\nUSN\tNAME\t\tBRANCH\t\tPHONE\t\tSEMESTER\n");
                while(temp->link!=NULL){
                        count++;
                        printf(\
                                "%s\t%s\t\t%s\t\t%llu\t\t%d\n", \
                                temp->usn,temp->name,temp->branch,temp->phno,temp->sem \
                        );
                        temp=temp->link;
                }
                printf("\nStudent count is %d\n",count);
        }
}

NODE insert_end(){
        NODE last=FIRST;
        printf("Enter the details of the students :\n");
        read();
        if(FIRST==NULL)
                FIRST=temp;
        else{
                while(last->link!=NULL)
                        last=last->link;
                last->link=temp;
        }
        return FIRST;
}

void delete_front(){
        temp=FIRST;
        if(FIRST==NULL)
                printf("List is empty !\n");
        else{
                printf("Deleted item is %s\n",temp->usn);
                FIRST=FIRST->link;
                free(temp);
        }
}

void delete_end(){
        NODE last=NULL;
        temp=FIRST;
        if(FIRST==NULL)
                printf("List is empty !\n");
        else if(FIRST->link==NULL){
                printf("Deleted item is %s\n",temp->usn);
                free(FIRST);
                FIRST=NULL;
        }else{
                while(temp->link!=NULL){
                        last=temp;
                        temp=temp->link;
                }
                last->link=NULL;
                printf("Deleted element is %s\n",temp->usn);
                free(temp);
        }
}

void main(){
        int choice;
        printf(\
                "\n1.Create SSL   \n2.Display SSL   \n3.Insert front \n4.Insert end \
                 \n5.Delete front \n6.Delete End   \n7.Exit" \
        );
        while(1){
                printf("\n> ");
                scanf("%d",&choice);
                switch(choice){
                        case 1:
                                create_SSL();
                                break;
                        case 2:
                                display_count();
                                break;
                        case 3:
                                printf("Enter the details of the students\n");
                                insert_front();
                                break;
                        case 4:
                                insert_end();
                                break;
                        case 5:
                                delete_front();
                                break;
                        case 6:
                                delete_end();
                                break;
                        case 7:
                                return;
                        default:
                                printf("Invalid choice !\n");
                }
        }
}
```