

**Design, develop and Implement a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Employee Data with the fields: SSN, Name, Dept, Designation, Sal, and PhNo**

- a. Create a DLL of N Employees Data by using end insertion.**
- b. Display the status of DLL and count the number of nodes in it**
- c. Perform Insertion and Deletion at End of DLL**
- d. Perform Insertion and Deletion at Front of DLL**
- e. Demonstrate how this DLL can be used as Double Ended Queue. f. Exit**

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

```
struct node
{
int ssn;
char name[10];
char dept[10];
char desg[10];
int sal;
int phno;

struct node *left;
struct node *right;
}; typedef struct node * NODE;
```

**//\*\*\*\*\* Function to Insert at the End of List \*\*\*\*\***

```
NODE insend(NODE first)
{
NODE newnode,temp;
newnode = (NODE) malloc(sizeof(struct node));

printf(" Enter the SSN Name Dept desg salary Phone No of student \n");
scanf("%d%s%s%s%d", &newnode->ssn, newnode->name, newnode->dept,
newnode->desg, &newnode->sal, &newnode->phno);
newnode->right = NULL;

if(first == NULL)
{
newnode->left = NULL;
first = newnode;
return first;
}

temp = first;
while(temp->right != NULL)
temp = temp->right;

temp->right = newnode;
newnode->left = temp;
return first;
}
```

**//\*\*\*\*\* Function to Delete Node from begining \*\*\*\*\***

```
NODE delfront(NODE first)
{
    NODE temp;

    if( first == NULL)
        printf(" The List is Empty, deletion cannot be possible\n");
    else
    {
        temp = first;
        first = first ->right;
        first->left = NULL;
        free(temp);
    }
    return first;
}
```

**//\*\*\*\*\* Function to Delete Node at End of List \*\*\*\*\***

```
NODE delend(NODE first)
{
    NODE prev,pres;

    if( first == NULL)
    {
        printf(" The List is Empty, deletion cannot be possible\n");
        return first;
    }

    if(first->right == NULL)
    {
        pres = first;
        first = NULL;
        free (pres);
        return first;
    }

    pres = first;

    while(pres->right != NULL)
    {
        prev = pres;
        pres = pres->right;
    }

    prev->right = NULL;
    free(pres);
    return first;
}
```

**//\*\*\*\*\* Function to Delete Node from begining \*\*\*\*\***

```
void display(NODE first)
{
    NODE temp;

    if( first == NULL)
        printf(" The List is Empty\n");
    else
    {
        temp = first;
        printf(" Enter the SSN Name Dept desg salary Phone No of student \n");

        while(temp != NULL )
        {
            printf("%d\t%s\t%s\t%s\t%d\t%d\n", temp->:ssn, temp->name, temp->dept,
                temp->desg, temp->sal, temp->phno);

            temp = temp->right;
        }
    }
}
```

**//main program**

```
void main()
{
    NODE first = NULL;
    int ch;
    clrscr();

    for(;;)
    {
        printf(" 1:Ins End 2:Del Front 4:Del end 5: Display\n");
        scanf("%d",&ch);

        switch(ch)
        {
            case 1: first = insend(first);
                    break;
            case 2: first = delfront(first);
                    break;
            case 3: first = delend(first);
                    break;
            case 4: display(first);
                    break;
            default: exit(0);
        }
    }
}
```