/\* Design, Develop and Implement a menu driven program in C for the following operation on Doubly linked list(DLL)

of professor data with the fieldss: ID, Name,Branch,Area of specialization

a)Create a DLL stack of N Professors Data

b) Create a DLL queue of N Professors Data

c) Display the status of DLL and count the number of nodes in it.

\*/

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

struct node

{

struct node\* prev;

int id;

char name[20], branch[20], area[20];

struct node\* next;

}\*head=NULL,\*newnode;

int count=0;

void create()

{

int id,n,i;

char name[20],branch[20],area[20];

newnode=(struct node \*)malloc(sizeof(struct node));

newnode->next=NULL;

newnode->prev=NULL;

printf("Enter the professor details\n");

printf("\n ID: ");

scanf("%d",&id);

printf("\n Name: ");

scanf("%s",name);

printf("\n Branch: ");

scanf("%s",branch);

printf("\n Area: ");

scanf("%s",area);

newnode->id=id;

strcpy(newnode->name,name);

strcpy(newnode->branch,branch);

strcpy(newnode->area,area);

}

void insert\_end()

{

struct node\* temp;

int n,i;

printf("Enter the number of professors\n");

scanf("%d",&n);

for(i=0;i<n;i++)

{

create();

printf("Node created");

if(head==NULL)

{

printf("%d",head);

newnode->next=head;

head=newnode;

head->prev=newnode;

}

else

{

temp=head;

while(temp->next!=NULL)

temp=temp->next;

temp->next=newnode;

newnode->prev=temp;

}

}

}

void delete\_end()

{

struct node\* temp,\*prevnode;

temp=head;

if(temp->next==NULL)

{

free(temp);

head=NULL;

}

else

{

while(temp->next!=NULL)

{

prevnode=temp;

temp=temp->next;

}

printf("The deleted data is\n");

printf("ID: %d\nName: %s\nBranch: %s\nArea:%s\n",temp->id,temp->name,temp->branch,temp->area);

free(temp);

prevnode->next=NULL;

}

}

void delete\_front()

{

struct node\* temp;

temp=head;

if(temp->next==NULL)

{

free(temp);

head=NULL;

}

else

{

head=temp->next;

printf("ID: %d\nName:%s\n,Branch:%s\n,Area:%s\n",temp->id,temp->name,temp->branch,temp->area);

free(temp);

}

}

void display()

{

struct node\*temp=head;

if(head==NULL)

{

printf("Empty\n");

return;

}

count=0;

printf("The professor details..\n");

while(temp!=NULL)

{

printf("ID:%d\nName:%s\nBranch:%s\nArea:%s\n",temp->id,temp->name,temp->branch,temp->area);

temp=temp->next;

count++;

}

printf("The number of nodes: %d\n",count);

}

int main()

{

int ch=1,op=1,n,i;

while(ch)

{

printf(" 1. create a DLL stack of N professors data\n");

printf(" 2. create a DLL queue of N professors data\n");

printf(" 3. Display the status with number of nodes in it\n");

printf(" 4. exit\n");

printf("Enter your choice\n");

scanf("%d",&ch);

switch(ch)

{

case 1: while(op)

{

printf("Stack of N professors\n");

printf("1.push the data\n");

printf("2.pop the data\n");

printf(" Enter 0 to exit\n");

printf("Enter your option\n");

scanf("%d",&op);

switch(op)

{

case 1:insert\_end();

break;

case 2: delete\_end();

break;

default: printf("Enter the correct choice\n");

break;

}

}

op=1;

break;

case 2: while(op)

{

printf("Queue of professors\n");

printf("1. Insert to the queue\n");

printf("2. Delete from the queue\n");

printf("Enter 0 to exit\n");

printf("Enter your choice\n");

scanf("%d",&op);

switch(op)

{

case 1: insert\_end();

break;

case 2: delete\_front();

break;

default: printf("Enter the correct choice\n");

break;

}

}

op=1;

break;

case 3: printf("Displaying...\n");

display();

break;

case 4: exit(1);

default: printf("Enter the correct choice\n");

break;

}

}

return 0;

}