/\*Name: Akansha Pankaj Sonar

Div:3C

Roll no:C23373

Subject: DSL

LAB ASSIGNMENT 7

Department of Computer Engineering has student's club named 'Pinnacle Club'. Students of Second, third and final year of department can be granted membership on request.

Similarly one may cancel the membership of club. First node is reserved for president of club and last node is reserved for secretary of club.

Write C++ program to maintain club memberâ€˜s information using singly linked list.

Store student PRN and Name. Write functions to

a) Add and delete the members as well as president or even secretary.

b) Compute total number of members of club

c) Display members

d) Two linked lists exists for two divisions. Concatenate two lists\*/

#include<iostream>

#include<string.h>

using namespace std;

struct node

{

int prn,rollno;

char name[50];

struct node \*next;

};

class info

{ node \*s=NULL,\*head1=NULL,\*temp1=NULL,\*head2=NULL,\*temp2=NULL,\*head=NULL,\*temp=NULL;

int b,c,i,j,ct;

char a[20];

public:

node \*create();

void insertp();

void insertm();

void delm();

void delp();

void dels();

void display();

void count();

void concat();

} ;

node \*info::create()

{ node \*p=new(struct node);

cout<<"enter name of student \n";

cin>>a;

strcpy(p->name,a);

cout<<"\n enter prn no. of student \n";

cin>>b;

p->prn=b;

cout<<"enter student rollno \n";

cin>>c;

p->rollno=c;

p->next=NULL;

return p;

}

void info::insertm()

{

node \*p=create();

if(head==NULL)

{

head=p;

}

else

{

temp=head;

while(temp->next!=NULL)

{

temp=temp->next;

}

temp->next=p;

}

}

void info::insertp()

{

node \*p=create();

if(head==NULL)

{

head=p;

}

else

{

temp=head;

head=p;

head->next=temp->next;

}

}

void info::display()

{

if(head==NULL)

{

cout<<"linklist is empty \n";

}

else

{

temp=head;

cout<<" prn rolln0 NAME \n";

while(temp->next!=NULL)

{

cout<<" \n"<<temp->prn<<" "<<temp->rollno<<" "<<temp->name;

temp=temp->next;

}

cout<<" "<<temp->prn<<" "<<temp->rollno<<" "<<temp->name;

}

}

void info::delm()

{ int m,f=0;

cout<<"\n enter the prn no. of student whose data you want to delete \n";

cin>>m;

temp=head;

while(temp->next!=NULL)

{

if(temp->prn==m)

{

s->next=temp->next;

delete(temp);

f=1;

}

s=temp;

temp=temp->next;

}

if(f==0)

{

cout<<"\n sorry memeber not deleted \n";

}

}

void info::delp()

{

temp=head;

head=head->next;

delete(temp);

}

void info::dels()

{

temp=head;

while(temp->next!=NULL)

{

s=temp;

temp=temp->next;

}

s->next=NULL;

delete(temp);

}

void info::count()

{

temp=head; ct=0;

while(temp->next!=NULL)

{

temp=temp->next; ct++;

}

ct++;

cout<<" Count of members is: \n"<<ct;

}

void info::concat()

{

int k,j;

cout<<"enter no. of members in list1 \n";

cin>>k;

head=NULL;

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

{

insertm();

head1=head;

}

head=NULL;

cout<<"enter no. of members in list2 \n";

cin>>j;

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

{

insertm();

head2=head;

}

head=NULL;

temp1=head1;

while(temp1->next!=NULL)

{

temp1=temp1->next;

}

temp1->next=head;

temp2=head1;

cout<<" PRN ROLL\_NO NAME \n";

while(temp2->next!=NULL)

{

cout<<"\n "<<temp2->prn<<" "<<temp2->rollno<<" "<<temp2->name<<"\n";;

temp2=temp2->next;

}

cout<<"\n "<<temp2->prn<<" "<<temp2->rollno<<" "<<temp2->name<<"\n";

}

int main()

{

info s;

int i;

char ch;

do

{

cout<<"\n choice the options ";

cout<<"\n 1. To insert president ";

cout<<"\n 2. To insert member ";

cout<<"\n 3. To insert secretary";

cout<<"\n 4. To delete president ";

cout<<"\n 5. To delete member ";

cout<<"\n 6. To delete secretary";

cout<<"\n 7. To display data ";

cout<<"\n 8. Count of members ";

cout<<"\n 9.To concatenate two strings \n ";

cin>>i;

switch(i)

{

case 1: s.insertp();

break;

case 2: s.insertm();

break;

case 3: s.insertm();

break;

case 4: s.delp();

break;

case 5: s.delm();

break;

case 6: s.dels();

break;

case 7: s.display();

break;

case 8: s.count();

break;

case 9: s.concat();

break;

default: cout<<"\n unknown choice \n";

}

cout<<"\n do you want to continue enter y/Y \n";

cin>>ch;

}

while(ch=='y'||ch=='Y');

return 0;

}