INPUT :

#include <iostream>

#include <cstring>

using namespace std;

// Data structure for student information, i.e., node

struct node {

int prn;

char name[20];

node \*next;

};

class panclub {

int num, cnt;

char nm[20]; // Data members

node \*head; // pointing to the first node

public:

panclub() { // Constructor to initialize object

num = cnt = 0;

head = nullptr;

}

node \*create();

void display(node \*);

node \*concat(node \*, node \*); // Member Functions with arguments

void reverse(node \*);

node \*insert\_president(node \*);

void insert\_sec(node \*);

void insert\_member(node \*);

node \*del\_president(node \*);

node \*del\_secretary(node \*);

node \*del\_member(node \*);

~panclub(); // Destructor to clean up memory

};

// Destructor to delete the entire linked list

panclub::~panclub() {

node \*current = head;

node \*nextNode;

while (current != nullptr) {

nextNode = current->next;

delete current;

current = nextNode;

}

}

// To create the list of Divisions

node\* panclub::create() {

node \*temp, \*n1;

temp = n1 = nullptr;

cout << "\nHow many students' data do you want to insert in the panclub database: ";

cin >> cnt;

while (cnt > 0) {

n1 = new node; // Allocate memory for all fields of struct

cout << "\nEnter the PRN number of student: ";

cin >> num;

n1->prn = num; // Storing the PRN in node field prn

cout << "\nEnter the name of the student: ";

cin >> nm;

strcpy(n1->name, nm); // Storing the name in node field name

n1->next = nullptr; // Making the next field null

if (head == nullptr) { // Check if head is empty

head = n1; // Make new node as head

temp = head;

} else {

temp = head;

while (temp->next != nullptr) { // Attach at the end of list

temp = temp->next;

}

temp->next = n1;

}

cnt--;

}

return head;

}

void panclub::display(node \*head) { // Display the list of both divisions

node \*temp;

temp = head;

while (temp != nullptr) {

if (temp->next == nullptr) {

cout << "[" << temp->prn << "|" << temp->name << "]->NULL";

} else {

cout << "[" << temp->prn << "|" << temp->name << "]->";

}

temp = temp->next;

}

cout << endl;

}

node\* panclub::concat(node \*head1, node \*head2) { // To concatenate both the divisions' data into one list

if (head1 == nullptr) return head2;

if (head2 == nullptr) return head1;

node \*temp = head1;

while (temp->next != nullptr) {

temp = temp->next;

}

temp->next = head2;

return head1;

}

void panclub::reverse(node \*head) {

if (head == nullptr) return;

reverse(head->next);

cout << "[" << head->prn << "|" << head->name << "]->";

}

node\* panclub::insert\_president(node \*head) {

node \*n2 = new node;

cout << "\nEnter the PRN number of the President: ";

cin >> n2->prn;

cout << "\nEnter the name of the President: ";

cin >> n2->name;

n2->next = head;

head = n2;

return head;

}

void panclub::insert\_member(node \*head) {

node \*temp = head, \*n2 = new node;

int pn;

cout << "\nEnter the PRN number of the Member: ";

cin >> n2->prn;

cout << "\nEnter the name of the Member: ";

cin >> n2->name;

n2->next = nullptr;

cout << "\nEnter the PRN number after which you want to add this member: ";

cin >> pn;

while (temp != nullptr) {

if (temp->prn == pn) {

n2->next = temp->next;

temp->next = n2;

break;

}

temp = temp->next;

}

cout << "\n\nMember added successfully!";

}

void panclub::insert\_sec(node \*head) {

node \*temp = head, \*n2 = new node;

cout << "\nEnter the PRN number of the Secretary: ";

cin >> n2->prn;

cout << "\nEnter the Name of the Secretary: ";

cin >> n2->name;

n2->next = nullptr;

while (temp->next != nullptr) {

temp = temp->next;

}

temp->next = n2;

}

// Delete the president node from the list

node\* panclub::del\_president(node \*head) {

if (head == nullptr) return nullptr;

node \*temp = head;

head = temp->next;

delete temp;

return head;

}

// Delete the secretary node from the list

node\* panclub::del\_secretary(node \*head) {

if (head == nullptr) return nullptr;

node \*temp = head, \*t1 = nullptr;

while (temp->next != nullptr) {

t1 = temp;

temp = temp->next;

}

if (t1 != nullptr) {

t1->next = nullptr;

}

delete temp;

return head;

}

// Delete the member from the list

node\* panclub::del\_member(node \*head) {

if (head == nullptr) return nullptr;

node \*temp = head, \*t1 = nullptr;

int pn;

cout << "\nEnter the PRN number after which you want to delete the member: ";

cin >> pn;

while (temp != nullptr) {

if (temp->prn == pn) {

t1 = temp->next;

if (t1 != nullptr) {

temp->next = t1->next;

delete t1;

}

break;

}

temp = temp->next;

}

cout << "\n\nMember removed successfully!";

return head;

}

int main() {

panclub p1, p2, p3;

node \*h1 = nullptr, \*h2 = nullptr, \*h3 = nullptr;

int ch;

cout << "\n\t!!! Group B: Assignment No: 01 !!!" << endl; // prints assignment number and group

do {

cout << "\n\n1. Enter data of SE A Division:";

cout << "\n2. Enter data of SE B Division:";

cout << "\n3. Concatenation of Lists..";

cout << "\n4. Exit.";

cout << "\nEnter your choice: ";

cin >> ch;

switch (ch) {

case 1:

cout << "\n\nPlease enter the student info who is a registered member..";

cout << "\n\nEnter the Panclub Data of SE A Division:\n";

h1 = p1.create();

cout << "\nSE Comp Division A List is as follows:\n\n";

p1.display(h1);

cout << "\n\nReverse List of SE Div A:\n\n";

p1.reverse(h1);

cout << "NULL\n";

p1.insert\_sec(h1);

cout << "\nAfter insertion of Secretary:\n";

p1.display(h1);

h1 = p1.insert\_president(h1);

cout << "\nAfter insertion of President:\n";

p1.display(h1);

p1.insert\_member(h1);

cout << "\nAfter insertion of member:\n";

p1.display(h1);

h1 = p1.del\_president(h1);

cout << "\n\nAfter deletion of president:\n";

p1.display(h1);

h1 = p1.del\_secretary(h1);

cout << "\n\nAfter deletion of secretary:\n";

p1.display(h1);

h1 = p1.del\_member(h1);

cout << "\n\nAfter deletion of member:\n";

p1.display(h1);

break;

case 2:

cout << "\n\nEnter the Panclub Data of SE B Division:\n";

h2 = p2.create();

cout << "\nSE Comp Division B List is as follows:\n\n";

p2.display(h2);

cout << "\n\nReverse List of SE Div B:\n\n";

p2.reverse(h2);

cout << "NULL\n";

p2.insert\_sec(h2);

cout << "\nAfter insertion of Secretary:\n";

p2.display(h2);

h2 = p2.insert\_president(h2);

cout << "\nAfter insertion of President:\n";

p2.display(h2);

p2.insert\_member(h2);

cout << "\nAfter insertion of member:\n";

p2.display(h2);

h2 = p2.del\_president(h2);

cout << "\n\nAfter deletion of president:\n";

p2.display(h2);

h2 = p2.del\_secretary(h2);

cout << "\n\nAfter deletion of secretary:\n";

p2.display(h2);

h2 = p2.del\_member(h2);

cout << "\n\nAfter deletion of member:\n";

p2.display(h2);

break;

case 3:

h3 = p3.concat(h1, h2);

cout << "\n\nThe concatenation of Div A and Div B of SE Comp Class is as follows:\n\n";

p3.display(h3);

break;

}

} while (ch != 4);

return 0;

}

--------------------------------------------------------------------------------------------------------------------------------------

OUTPUT:

1. Enter data of SE A Division:

2. Enter data of SE B Division:

3. Concatenation of Lists..

4. Exit.

Enter your choice: 1

Please enter the student info who is a registered member..

Enter the Panclub Data of SE A Division:

How many students' data do you want to insert in the panclub database: 2

Enter the PRN number of student: 123

Enter the name of the student: Abhishek

Enter the PRN number of student: 321

Enter the name of the student: Pawan

SE Comp Division A List is as follows:

[123|Abhishek]->[321|Pawan]->NULL

Reverse List of SE Div A:

[321|Pawan]->[123|Abhishek]->NULL

Enter the PRN number of the Secretary: 456

Enter the Name of the Secretary: Mayur

After insertion of Secretary:

[123|Abhishek]->[321|Pawan]->[456|Mayur]->NULL

Enter the PRN number of the President: 567

Enter the name of the President: Mohit

After insertion of President:

[567|Mohit]->[123|Abhishek]->[321|Pawan]->[456|Mayur]->NULL

Enter the PRN number of the Member: 696

Enter the name of the Member: Bhavesh

Enter the PRN number after which you want to add this member: 321

Member added successfully!

After insertion of member:

[567|Mohit]->[123|Abhishek]->[321|Pawan]->[696|Bhavesh]->[456|Mayur]->NULL

After deletion of president:

[123|Abhishek]->[321|Pawan]->[696|Bhavesh]->[456|Mayur]->NULL

After deletion of secretary:

[123|Abhishek]->[321|Pawan]->[696|Bhavesh]->NULL

Enter the PRN number after which you want to delete the member: 567

Member removed successfully!

After deletion of member:

[123|Abhishek]->[321|Pawan]->[696|Bhavesh]->NULL

1. Enter data of SE A Division:

2. Enter data of SE B Division:

3. Concatenation of Lists..

4. Exit.

Enter your choice: 3

The concatenation of Div A and Div B of SE Comp Class is as follows:

[123|Abhishek]->[321|Pawan]->[696|Bhavesh]->NULL

1. Enter data of SE A Division:

2. Enter data of SE B Division:

3. Concatenation of Lists..

4. Exit.

Enter your choice: 4