Assignment . 9

#include <iostream>

using namespace std;

struct Node {

int seatNumber;

bool isBooked;

Node\* prev;

Node\* next;

};

class Row {

public:

Node\* head;

Row() {

head = nullptr;

for (int i = 1; i <= 7; i++) {

Node\* newNode = new Node();

newNode->seatNumber = i;

newNode->isBooked = false;

if (!head) {

head = newNode;

head->next = head;

head->prev = head;

} else {

Node\* tail = head->prev;

tail->next = newNode;

newNode->prev = tail;

newNode->next = head;

head->prev = newNode;

}

}

}

void displaySeats() {

Node\* temp = head;

do {

cout << "Seat " << temp->seatNumber << ": " << (temp->isBooked ? "Booked" : "Available") << endl;

temp = temp->next;

} while (temp != head);

}

bool bookSeat(int seatNumber) {

Node\* temp = head;

do {

if (temp->seatNumber == seatNumber && !temp->isBooked) {

temp->isBooked = true;

return true;

}

temp = temp->next;

} while (temp != head);

return false;

}

bool cancelBooking(int seatNumber) {

Node\* temp = head;

do {

if (temp->seatNumber == seatNumber && temp->isBooked) {

temp->isBooked = false;

return true;

}

temp = temp->next;

} while (temp != head);

return false;

}

};

int main() {

Row\* rows[10];

for (int i = 0; i < 10; i++) {

rows[i] = new Row();

}

int choice, rowNumber, seatNumber;

do {

cout << "\n1. Display available seats\n2. Book a seat\n3. Cancel a booking\n4. Exit\n";

cout << "Enter your choice: ";

cin >> choice;

switch (choice) {

case 1:

cout << "Enter row number (1-10): ";

cin >> rowNumber;

if (rowNumber >= 1 && rowNumber <= 10) {

rows[rowNumber - 1]->displaySeats();

} else {

cout << "Invalid row number!" << endl;

}

break;

case 2:

cout << "Enter row number (1-10): ";

cin >> rowNumber;

cout << "Enter seat number (1-7): ";

cin >> seatNumber;

if (rowNumber >= 1 && rowNumber <= 10 && seatNumber >= 1 && seatNumber <= 7) {

if (rows[rowNumber - 1]->bookSeat(seatNumber)) {

cout << "Seat " << seatNumber << " in row " << rowNumber << " booked successfully." << endl;

} else {

cout << "Seat " << seatNumber << " in row " << rowNumber << " is already booked." << endl;

}

} else {

cout << "Invalid row or seat number!" << endl;

}

break;

case 3:

cout << "Enter row number (1-10): ";

cin >> rowNumber;

cout << "Enter seat number (1-7): ";

cin >> seatNumber;

if (rowNumber >= 1 && rowNumber <= 10 && seatNumber >= 1 && seatNumber <= 7) {

if (rows[rowNumber - 1]->cancelBooking(seatNumber)) {

}

}

Enter row number (1-10): 1

Seat 1: Available

Seat 2: Available

Seat 3: Available

Seat 4: Available

Seat 5: Available

Seat 6: Available

Seat 7: Available

Enter row number (1-10): 1

Seat 1: Available

Seat 2: Available

Seat 3: Booked

Seat 4: Available

Seat 5: Available

Seat 6: Available

Seat 7: Available