**ЗВІТ**

**до лабораторної роботи № < 9.3 >**

**« Опрацювання масиву структур »**

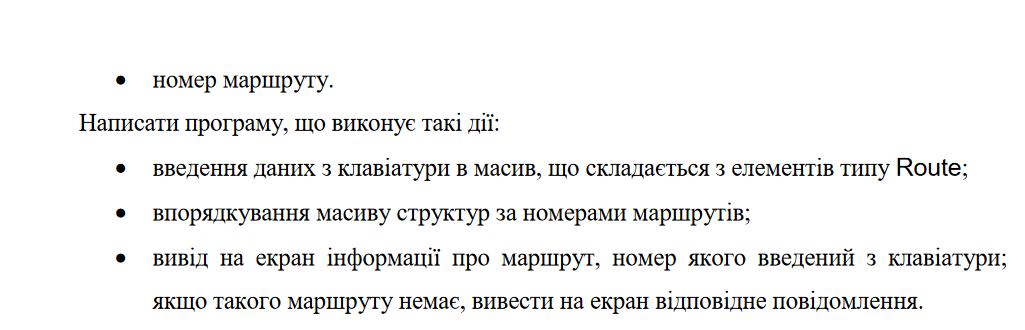
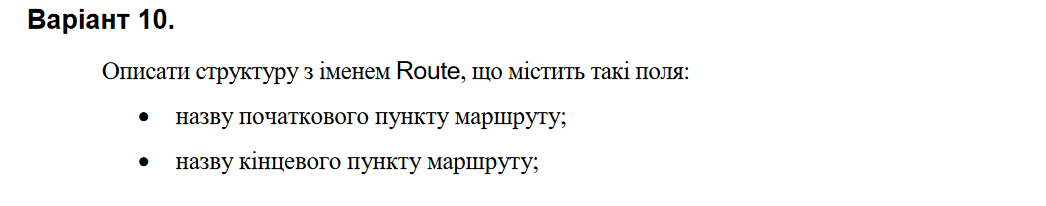
**з дисципліни**

**«Алгоритмізація та програмування»**

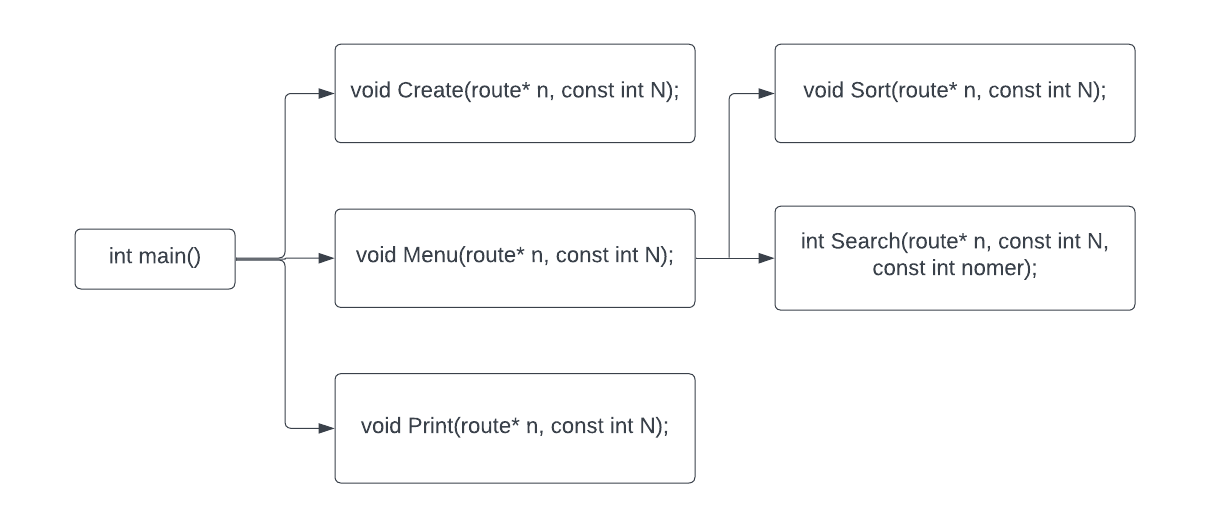
**студента групи ІН-105Б**

**Горанова Анастаса**

**Умова завдання:**



**Структурна схема:**



**Текст програми:**

// 9.3.cpp

// < Горанов Анастас >

// Лабораторна робота № 9.3

// Опрацювання масиву структур

// Варіант 10

#include <iostream>

#include <iomanip>

#include <string>

using namespace std;

struct route

{

string start;

string end;

int nomer;

};

void Create(route\* n, const int N);

void Print(route\* n, const int N);

void Menu(route\* n, const int N);

void Sort(route\* n, const int N);

int Search(route\* n, const int N, const int nomer);

int main() {

int N;

cout << "Enter the number of routes: "; cin >> N;

route\* n = new route[N];

Create(n, N);

Print(n, N);

Menu(n, N);

}

void Create(route\* n, const int N)

{

for (int i = 0; i < N; i++)

{

cout << "Route #" << i + 1 << ":" << endl;

cout << " Route number: "; cin >> n[i].nomer;

cin.get();

cin.sync();

cout << " Start: "; getline(cin, n[i].start);

cout << " End: "; getline(cin, n[i].end);

cout << endl;

}

}

void Print(route\* n, const int N)

{

cout << "====================================="

<< endl;

cout << "| # | Start | End | Number " << setw(4) << "|"

<< endl;

cout << "-------------------------------------"

<< endl;

for (int i = 0; i < N; i++)

{

cout << "| " << setw(1) << right << i + 1 << " ";

cout << "| " << setw(7) << left << n[i].start << " "

<< "| " << setw(6) << right << n[i].end << " "

<< "| " << setw(10) << left << n[i].nomer << "|" << endl;

}

cout << "=====================================" << endl << endl;

}

void Menu(route\* n, const int N) {

int choice;

do {

cout << "--------------------------\n";

cout << "MENU\n";

cout << "[1] - Sort\n";

cout << "[2] - Search\n";

cout << "[3] - Exit\n";

cout << "--------------------------\n";

cout << "Choose: ";

cin >> choice;

switch (choice) {

case 1:

cout << "Sorted:\n";

Sort(n, N);

Print(n, N);

break;

case 2:

{

cout << "Search:\n";

int number, found;

cout << "Enter route number: ";

cin >> number;

found = Search(n, N, number);

if (found != -1)

{

cout << "Start: " << n[found].start << endl;

cout << "End: " << n[found].end << endl;

}

else

cout << "No route with this number was found." << endl;

}

case 3:

break;

default:

cout << "--------------------------\n";

cout << "MENU\n";

cout << "[1] - Sort\n";

cout << "[2] - Search\n";

cout << "[3] - Exit\n";

cout << "--------------------------\n";

cout << "Choose:";

cin >> choice;

}

} while (choice != 3);

}

void Sort(route\* n, const int N) {

route tmp;

for (int i0 = 0; i0 < N - 1; i0++) {

for (int i1 = 0; i1 < N - i0 - 1; i1++) {

if (n[i1].nomer > n[i1 + 1].nomer)

{

tmp = n[i1];

n[i1] = n[i1 + 1];

n[i1 + 1] = tmp;

}

}

}

}

int Search(route\* n, const int N, const int nomer) {

for (int i = 0; i < N; i++)

if (n[i].nomer == nomer)

return i;

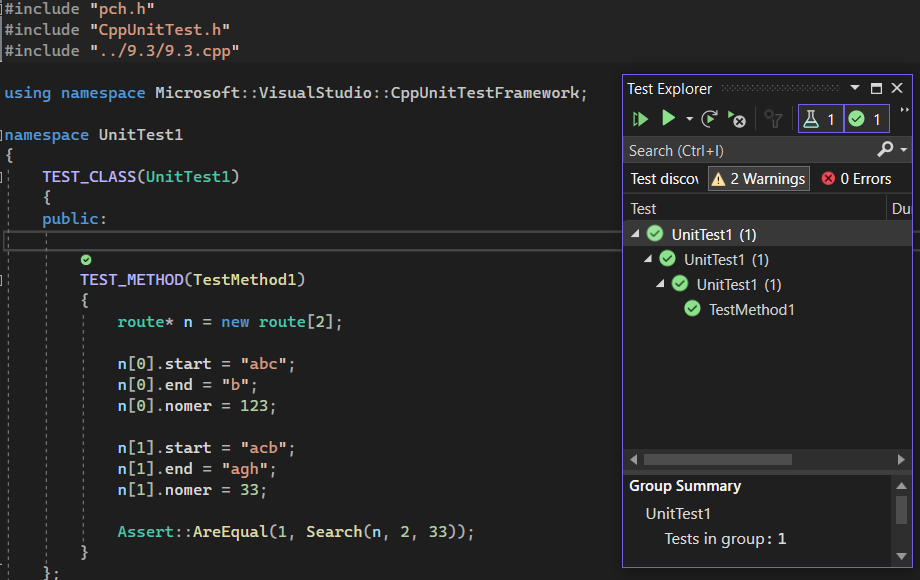
return -1;

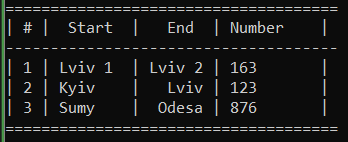
}

**Посилання на git-репозиторій з проектом:**

*https://github.com/StassNG/9.3*

**Текст та результати unit-тесту:**

****

****

**Висновки:** на цій лабораторній роботі я навчився опрацьовувати масиви структур.