МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 11

з навчальної дисципліни

“Базові методології та технології програмування”

РЕАЛІЗАЦІЯ ПРОГРАМНИХ ЗАСОБІВ ОБРОБЛЕННЯ ДИНАМІЧНИХ СТРУКТУР ДАНИХ ТА БІНАРНИХ ФАЙЛІВ

ВИКОНАВ

студент академічної групи КБ-23

\_\_\_\_\_\_\_\_ Назар

КОКІШ

ПЕРЕВІРИЛА

викладач кафедри кібербезпеки

та програмного забезпечення

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Ганна ДРЄЄВА

Кропивницький – 2023

**ТЕМА: РЕАЛІЗАЦІЯ ПРОГРАМНИХ ЗАСОБІВ ОБРОБЛЕННЯ ДИНАМІЧНИХ СТРУКТУР ДАНИХ ТА БІНАРНИХ ФАЙЛІВ**

**МЕТА:** Набути ґрунтовних вмінь і практичних навичок командної (колективної) реалізації програмного забезпечення, розроблення функцій оброблення динамічних структур даних, використання стандартних засобів С++ для керування динамічною пам’яттю та бінарними файловими потоками.

**ЗАВДАННЯ:**

1. У складі команди ІТ-проєкту розробити програмні модулі оброблення динамічної структури даних.

2. Реалізувати програмний засіб на основі розроблених командою ІТ-проєкту модулів.

**Завдання на розроблення ПЗ:**

Створити інформаційну систему "Розклад руху автобусів, АС Кропивницький" (номер рейсу та маршрут, періодичність / щоденно, парні, непарні, день або дні тижня/ ,час прибуття /гг:хх/, час прибуття до кінцевої станції /гг:хх/).

За вибором оператора програма забезпечує:

- пошук запису в ІС за введеною назвою початкової або кінцевої автостанції (без розрізнення регістра, виводяться усі знайдені рейси)

- виведення всього розкладу руху автобусів з ІС на екран

- додавання нового рейса до ІС

- вилучення заданого рейса з ІС (за номером)

- завершення роботи програми з автоматичним записом ІС у файл

Дані ІС автоматично завантажуються з файлу під час запуску програми

**Лістинг структури:**

#pragma once

#include <string>

using namespace std;

struct BusSchedule {

int number;

string route;

string frequency;

string days;

string arrivalTime;

string finalStationArrivalTime;

};

**Лістинг модуля prj\_3\_Konstantynov.h:**

#pragma once

#include <vector>

#include "struct\_type\_priject\_3.h"

#include <iostream>

using namespace std;

void searchByStation(const vector<BusSchedule>& schedule, const string& stationName) {

cout << "Рейси з автостанції " << stationName << ":" << endl;

for (const auto& bus : schedule) {

if (bus.route.find(stationName) != string::npos) {

cout << "Номер рейсу: " << bus.number << ", Маршрут: " << bus.route << ", Час прибуття: " << bus.arrivalTime << endl;

}

}

}

void printSchedule(const vector<BusSchedule>& schedule) {

cout << "Розклад руху автобусів:" << endl;

for (const auto& bus : schedule) {

cout << "Номер рейсу: " << bus.number << ", Маршрут: " << bus.route << ", Періодичність: " << bus.frequency << ", Дні тижня: " << bus.days << ", Час прибуття: " << bus.arrivalTime << ", Час прибуття до кінцевої станції: " << bus.finalStationArrivalTime << endl;

}

}

**Лістинг модуля prj\_3\_Kokish.h:**

#pragma once

#include "struct\_type\_priject\_3.h"

#include <iostream>

#include <vector>

using namespace std;

void addNewBus(vector<BusSchedule>& schedule) {

BusSchedule newBus;

cout << "Введіть номер рейсу: ";

cin >> newBus.number;

cout << "Введіть маршрут: ";

cin.ignore();

getline(cin, newBus.route);

cout << "Введіть періодичність (щоденно/парні/непарні): ";

getline(cin, newBus.frequency);

cout << "Введіть дні тижня: ";

getline(cin, newBus.days);

cout << "Введіть час прибуття: ";

getline(cin, newBus.arrivalTime);

cout << "Введіть час прибуття до кінцевої станції: ";

getline(cin, newBus.finalStationArrivalTime);

schedule.push\_back(newBus);

cout << "Новий рейс успішно додано до розкладу." << endl;

}

void deleteBus(vector<BusSchedule>& schedule, int number) {

auto it = find\_if(schedule.begin(), schedule.end(), [number](const BusSchedule& bus) { return bus.number == number; });

if (it != schedule.end()) {

schedule.erase(it);

cout << "Рейс номер " << number << " успішно вилучено з розкладу." << endl;

}

else {

cout << "Рейс з номером " << number << " не знайдено у розкладі." << endl;

}

}

**Лістинг модуля prj\_3\_Lytvyn.h:**

#pragma once

#include "struct\_type\_priject\_3.h"

#include <iostream>

#include <vector>

#include <fstream>

using namespace std;

void saveScheduleToFile(const vector<BusSchedule>& schedule, const string& filename) {

ofstream file(filename);

if (!file.is\_open()) {

cout << "Помилка: Неможливо відкрити файл для запису." << endl;

return;

}

for (const auto& bus : schedule) {

file << bus.number << "," << bus.route << "," << bus.frequency << "," << bus.days << "," << bus.arrivalTime << "," << bus.finalStationArrivalTime << endl;

}

file.close();

}

**Лістинг програмного засобу:**

#include "prj\_3\_Konstantynov.h"

#include "prj\_3\_Kokish.h"

#include "prj\_3\_Lytvyn.h"

#include <Windows.h>

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

vector<BusSchedule> schedule;

string filename = "bus\_schedule.txt";

ifstream file(filename);

if (file.is\_open()) {

int number;

char comma;

while (file >> number) {

BusSchedule bus;

bus.number = number;

file >> comma >> ws;

getline(file, bus.route, ',');

getline(file, bus.frequency, ',');

getline(file, bus.days, ',');

getline(file, bus.arrivalTime, ',');

getline(file, bus.finalStationArrivalTime);

schedule.push\_back(bus);

}

file.close();

}

else {

cout << "Не вдалося завантажити розклад з файлу. Файл буде створено пізніше." << endl;

}

int choice;

do {

cout << "\nМеню:\n1. Пошук за назвою станції\n2. Виведення розкладу\n3. Додавання нового рейсу\n4. Вилучення рейсу\n5. Завершення роботи програми" << endl;

cout << "Оберіть опцію: ";

cin >> choice;

switch (choice) {

case 1: {

string stationName;

cout << "Введіть назву автостанції: ";

cin.ignore();

getline(cin, stationName);

searchByStation(schedule, stationName);

break;

}

case 2:

printSchedule(schedule);

break;

case 3:

addNewBus(schedule);

break;

case 4: {

int number;

cout << "Введіть номер рейсу для вилучення: ";

cin >> number;

deleteBus(schedule, number);

break;

}

case 5:

saveScheduleToFile(schedule, filename);

cout << "Дякую за користування програмою. До побачення!" << endl;

break;

default:

cout << "Невірний вибір. Будь ласка, виберіть опцію знову." << endl;

}

} while (choice != 5);

return 0;

}

Висновок. У ході виконання лабораторної роботи, мені довелося працювати з іншими студентами для виконання спільного проєкту. Під час роботи ми разом визначали архітектуру програми, аналізували поставлені задачі та шукали кращі проєктні рішення. У своєму модулі я розробляв функції для додавання нового рейса до ІС та вилучення заданого рейса з ІС (за номером), при роботі я використовував підключення написаної нами структури та бібліотеку векторів.