

Міністерство освіти і науки України  
Центральноукраїнський національний технічний університет  
Механіко-технологічний факультет  
Кафедра кібербезпеки та програмного забезпечення

Звіт  
Про виконання Лабораторії роботи №11  
з дисципліни  
“Базові методології та технології програмування”  
на тему  
«Реалізація програмних засобів оброблення динамічних структур  
даних та бінарних файлів»

Виконали

Студент академічної групи КН-22

Гончарук О.С.

Студент академічної групи КБ-22-1

Лук'яненко Р.О.

Студент академічної групи КБ-22-1

Паращенко Д.С.

Перевірів

викладач

Собінов О.Г

**Тема:** Реалізація програмних засобів оброблення динамічних структур даних та бінарних файлів.

**Мета роботи:** Набуття ґрунтовних вмінь і практичних навичок командної (колективної) реалізації програмного забезпечення, розроблення функцій оброблення динамічних структур даних, використання стандартних засобів C++ для керування динамічною пам'яттю та бінарними файловими потоками.

**Завдання до лабораторної роботи:**

1. У складі команди ІТ-проекта розробити програмні модулі оброблення динамічної структури даних.
2. Реалізувати програмний засіб на основі розроблених командою ІТ-проекта модулів.

## Варіант 2

<https://github.com/odorenskyi/Honcharuk-Oleksandr-KN22>

<https://github.com/odorenskyi/Lukianenko-Roman-KB221>

<https://github.com/odorenskyi/Parashchenko-Denys-KB221>

## ВАРІАНТ 2

### — ЗАВДАННЯ НА РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ —

Створити **електронний довідник телефонних кодів міст України** та мобільних операторів (назва міста, код\*)

За вибором користувача (оператора) програма забезпечує:

- пошук запису в довіднику за введеним кодом або назвою міста;
- виведення всього довідника на екран або у текстовий файл;
- додавання нового запису в довідник;
- вилучення заданого оператором запису з довідника;
- завершення роботи програми з автоматичним записом довідника у файл.

Дані довідника автоматично завантажуються з файлу під час запуску програми.

\*Коди телефонів областей України можуть бути від 3 до 5 цифр; з жовтня 2009 року дзвінки здійснюються через набір 0 (для міжміських дзвінків по Україні) та 00 (для міжнародних дзвінків з України).

| Місто           | Код  |
|-----------------|------|
| Олександрія     | 5235 |
| Балахівка       | 5237 |
| Бобринець       | 5257 |
| Гайворон        | 5254 |
| Голованівськ    | 5252 |
| Димитрове       | 5235 |
| Добровеличківка | 5253 |

|                 |      |
|-----------------|------|
| Долинська       | 5234 |
| Єлизаветградка  | 5242 |
| Завалля         | 5254 |
| Знам'янка       | 5233 |
| Знам'янка Друга | 5233 |
| Кропивницький   | 522  |
| Компаніївка     | 5240 |
| Мала Виска      | 5258 |
| Молодіжне       | 5234 |

|                  |      |
|------------------|------|
| Нова Прага       | 5235 |
| Новгородка       | 5241 |
| Новоархангельськ | 255  |
| Новомиргород     | 5256 |
| Новоукраїнка     | 5251 |
| Олександрівка    | 5242 |
| Олександрія      | 5235 |
| Ольшанка         | 5250 |
| Онуфріївка       | 5238 |

## Лістинги коду:

### struct\_type\_project\_2.hst:

```
#ifndef STRUCT_TYPE_PROJECT_2_H
```

```
#define STRUCT_TYPE_PROJECT_2_H
```

```
#include <iostream>
```

```
#include <fstream>
```

```
#include <map>
```

```
struct CityCode {
```

```
    std::string city;
```

```
    std::string code;
```

```
};
```

```
void saveDictionary(const std::map<std::string, std::string>& dict, const std::string& fileName);
```

```
void loadDictionary(std::map<std::string, std::string>& dict, const std::string& fileName);
```

```
void addRecord(std::map<std::string, std::string>& dict);
```

```
void deleteRecord(std::map<std::string, std::string>& dict);
```

```
void printDictionary(const std::map<std::string, std::string>& dict);
```

```
#endif // STRUCT_TYPE_PROJECT_2_H
```

**main.cpp:**

```
#include <iostream>
```

```
#include <fstream>
```

```
#include <map>
```

```
#include <windows.h>
```

```
#include "struct_type_project_2.h"
```

```
using namespace std;
```

```
void saveDictionary(const map<string, string>& dict, const string & fileName) {
```

```
    ofstream file(fileName);
```

```
    if (file.is_open()) {
```

```
        for (auto const& pair : dict) {
```

```
            string key = pair.first;
```

```
            string value = pair.second;
```

```
                file << key << "," << value << endl;
```

```
        }
```

```
        file.close();
```

```
    } else {
```

```
        cout << "Не удалось зберегти словник у файл" << endl;
```

```
    }
```

```
}
```

```
void loadDictionary(map<string, string>& dict, const string& fileName) {
```

```
    ifstream file(fileName);
```

```
if (file.is_open()) {  
    string line;  
    while (getline(file, line)) {  
        size_t pos = line.find(",");  
        if (pos != string::npos) {  
            string city = line.substr(0, pos);  
            string code = line.substr(pos + 1);  
            dict[city] = code;  
        }  
    }  
    file.close();  
} else {  
    cout << "Не вдалося завантажити словник із файлу" << endl;  
}  
}
```

```
void addRecord(map<string, string>& dict) {  
    string city, code;  
    cout << "Введіть назву міста: ";  
    getline(cin, city);  
    cout << "Введіть код: ";  
    getline(cin, code);  
    dict[city] = code;  
}
```

```
void deleteRecord(map<string, string>& dict) {  
    string city;  
    cout << "Введіть назву міста для видалення: ";  
    getline(cin, city);  
    auto it = dict.find(city);  
    if (it != dict.end()) {  
        dict.erase(it);  
        cout << "Запис видалено" << endl;  
    } else {  
        cout << "Запис не знайдено" << endl;  
    }  
}
```

```
void printDictionary(const map<string, string>& dict) {  
    for (auto const& pair : dict) {  
        string key = pair.first;  
        string value = pair.second;  
        cout << key << ": " << value << endl;  
    }  
}
```

```
int main() {  
    SetConsoleOutputCP(1251);  
    SetConsoleCP(1251);  
    map<string, string> dict;
```

```
loadDictionary(dict, "dictionary.txt");

cout << "Ласкаво просимо до словника телефонних кодів" << endl;

while (true) {

    cout << "Виберіть варіант:" << endl;

    cout << "1. Додати запис" << endl;

    cout << "2. Видалити запис" << endl;

    cout << "3. Вивести словник" << endl;

    cout << "4. Зберегти та вийти" << endl;

    int choice;

    cin >> choice;

    cin.ignore(); // ігнорувати символ нового рядка

    switch (choice) {

        case 1:

            addRecord(dict);

            break;

        case 2:

            deleteRecord(dict);

            break;

        case 3:

            printDictionary(dict);

            break;

        case 4:

            saveDictionary(dict, "dictionary.txt");

            cout << "Словник збережено. Вихід..." << endl;

            return 0;

    }
```



default:

```
cout << "Невірний вибір" << endl;
```

```
}
```

```
}
```

```
}
```

### TestSuite:

|  |   |
|--|---|
| Назва тестового набору<br>Test Suite Description | Лабораторна робота 11                                     |
| Назва проекту / ПЗ<br>Name of Project / Software | Лабораторна робота 11                                     |
| Рівень тестування<br>Level of Testing            | Системний<br>System Testing                               |
| Автор тест-сюита<br>Test Suite Author            | ІТ – команда:<br>Гончарук О., Лук'яненко.Р, Паращенко Д.. |
| Виконавець<br>Implementer                        | ІТ – команда  |

| Ід-р<br>Тест-<br>кейса /<br>Test<br>Case<br>ID | Дії (кроки) / Action (Test Steps)  | Очікуваний<br>результат / Exprected<br>Result                                 | Результат тестування<br>(пройшов/не<br>вдалося/заблокований) /<br>Test Result<br>(passed/failed/blocked) |
|--|--|---|--|
| 11.1   | <p>1. Запустити застосунок<br/>2. Обрати додавання нового<br/>запису<br/>-----<br/>Ласкаво просимо до словника<br/>телефонних кодів міст України!<br/>Виберіть варіант:<br/>1. Додати запис<br/>2. Видалити запис<br/>3. Вивести словник<br/>4. Зберегти та вийти<br/>1<br/>Введіть назву міста:<br/>Олександрія<br/>Введіть код: 5235<br/>Виберіть варіант:<br/>1. Додати запис<br/>2. Видалити запис</p> | <p>Зміст файла<br/>dictionary.txt<br/>Олександрія,5235<br/>Балахівка,5237</p> | passed   |

|      |  |   |        |
|------|--|---|--------|
|      | 3. Вивести словник<br>4. Зберегти та вийти<br>1<br>Введіть назву міста:<br>Балахівка<br>Введіть код: 5237  |   |        |
| 11.2 | 1. Запустити застосунок<br>2. Обрати виведення словника<br>-----<br>Ласкаво просимо до словника телефонних кодів міст України!<br>Виберіть варіант:<br>1. Додати запис<br>2. Видалити запис<br>3. Вивести словник<br>4. Зберегти та вийти<br>3   | Результат:<br>Олександрія: 5235<br>Балахівка: 5237              | passed |
| 11.3 | 1. Запустити застосунок<br>2. Обрати видалення запису<br>-----<br>Ласкаво просимо до словника телефонних кодів міст України!<br>Виберіть варіант:<br>1. Додати запис<br>2. Видалити запис<br>3. Вивести словник<br>4. Зберегти та вийти<br>2<br>Введіть назву міста для видалення: Олександрія | Результат:<br>Запис видалено<br>Зміст файлу:<br>Балахівка: 5237 | passed |
| 11.4 | 1. Запустити застосунок<br>2. Обрати вихід<br>-----<br>Ласкаво просимо до словника телефонних кодів міст України!<br>Виберіть варіант:<br>1. Додати запис<br>2. Видалити запис<br>3. Вивести словник<br>4. Зберегти та вийти<br>4  | Результат:<br>Словник збережено.<br>Вихід...                    | passed |

### **Висновок:**

Під час підготовки до виконання лабораторної роботи було належно опрацьовано рекомендовану літературу та контрольні запитання. Після чого було розпочато виконання лабораторної роботи.

Завантажений власний Git-репозиторій <https://github.com/odorenskyi/Honcharuk-Oleksandr-KN22>. У \Lab11 заповнено файл README.md, створено теки prj, Software, TestSuite, Report. До звіту включено тему мету завдання. Здійснено аналіз і постановку задачі. Виконано аналіз вимог, проектування архітектури, детальне проектування програмних модулів розв'язування задачі. Ми зібрались командою на мітинг та обговорювали специфікацію ПЗ, алгоритми вирішення її, проектні рішення. Потім кожен знає приступив до роботи, знаючи своє завдання. Командою розроблено набори контрольних прикладів до задачі задля виконання модульного тестування (Unit testing) модулів C++. За отриманими під час проектування програмних модулів артефактами командою виконано конструювання функцій: мовою програмування C++ реалізовано функції, які за наданим інтерфейсом реалізують розв'язування задачі.

В ході роботи над лабораторною роботою наша команда набула ґрунтовних вмінь і практичних навичок реалізації технології модульного програмування, застосування операторів C/C++, праці з файлами, розроблення статичних бібліотек, заголовкових файлів та програмних засобів у кросплатформовому середовищі Code::Blocks.

Процес виконання лабораторної роботи був зрозумілим та цікавим.