

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

ЗАВДАННЯ ВИДАВ  
доцент кафедри кібербезпеки  
та програмного забезпечення  
Доренський О. П.  
<https://github.com/odorenskyi/>

ВИКОНАЛИ  
студенти  
Гой Д. О. (КІ-22-1)  
Калашник О. В. (КН-22)  
Невгоденко В. І. (КІ-22-2)

ПЕРЕВІРИВ  
ст. викладач кафедри кібербезпеки  
та програмного забезпечення  
Собінов О. Г.

**Мета:** 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

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

<https://github.com/odorenskyi/Nevhodenko-Vadym-KI222>

<https://github.com/odorenskyi/Hoi-Dmytro-KI221>

## ВАРІАНТ 13

### — ЗАВДАННЯ НА РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ —

Реалізувати **електронний реєстр суб'єктів освітньої діяльності** (Код закладу в ЄДЕБО, Повне найменування, Коротка назва, Повне найменування (англ.), Форма власності, Найменування органу, до сфери управління якого належить заклад освіти, Найменування посади, прізвище, ім'я, по батькові керівника закладу освіти, Місцезнаходження (юридична адреса), Телефон / факс, Електронна пошта, Веб-сайт, ОЦ «Крим-Україна», ОЦ «Донбас-Україна», Військова кафедра, Рік заснування).

За вибором працівника поліції програма забезпечує:

- виведення всього реєстру на екран або у заданий текстовий файл;
- додавання нового запису до реєстру;
- пошук запису в реєстрі за кодом ЗВО в ЄДЕБО (якщо запис відсутній, виводиться відповідне повідомлення);
- вилучення заданого запису з реєстру;
- завершення роботи програми з автоматичним записом реєстру у файл.

Реєстр автоматично завантажується з файлу під час запуску програми.

РЕЄСТР СУБ'ЄКТІВ ОСВІТНЬОЇ ДІЯЛЬНОСТІ	
ЗАКЛАД ВИЩОЇ ОСВІТИ КРОПИВНИЦЬКИЙ	
ЦЕНТРАЛЬНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ	
Код закладу в ЄДЕБО	90
Повне найменування	Центральноукраїнський національний технічний університет
Коротка назва	ЦНТУ
Повне найменування (англ.)	Central Ukrainian National Technical University
Форма власності	Державна
Найменування органу, до сфери управління якого належить заклад освіти	Міністерство освіти і науки України
Найменування посади, прізвище, ім'я, по батькові керівника закладу освіти	Ректор Кропивний Володимир Миколайович
Місцезнаходження (юридична адреса)	Україна, 25006, Кіровоградська обл., м. Кропивницький, просп. Університетський, 8
Телефон / факс	+38(052)255-92-34
Електронна пошта	rector@kntu.kr.ua
Веб-сайт	http://www.kntu.kr.ua
ОЦ «Крим-Україна», ОЦ «Донбас-Україна»	так
Військова кафедра	ні
Рік заснування	1929

### Лістинг struct\_type\_project13.h:

```
#ifndef STRUCT_TYPE_PROJECT13_H_INCLUDED
#define STRUCT_TYPE_PROJECT13_H_INCLUDED

struct Subject {
    std::string code;
    std::string fullName;
    std::string shortName;
    std::string fullNameEng;
    std::string ownershipForm;
    std::string governingBody;
    std::string head;
    std::string location;
    std::string contact;
    std::string email;
    std::string website;
    bool hasCrimeaUkraineCenter;
    bool hasDonbasUkraineCenter;
    bool hasMilitaryDepartment;
    int foundationYear;
};

#endif // STRUCT_TYPE_PROJECT13_H_INCLUDED
```

### Лістинг lab11.cpp:

```
#include <iostream>
#include <fstream>
#include <vector>
#include <limits>
#include <windows.h>
#include "..\struct_type_project13.h"

void registry_from_file(Subject subject, std::vector<Subject>& registry)
{
    std::ifstream input("register.data");
    if (input.is_open())
    {
        while (input >> subject.code)
        {
            input.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
```

```

        std::getline(input, subject.fullName);
        std::getline(input, subject.shortName);
        std::getline(input, subject.fullNameEng);
        std::getline(input, subject.ownershipForm);
        std::getline(input, subject.governingBody);
        std::getline(input, subject.head);
        std::getline(input, subject.location);
        std::getline(input, subject.contact);
        std::getline(input, subject.email);
        std::getline(input, subject.website);
        std::string hasCrimeaUkraineCenterInput;
        std::getline(input, hasCrimeaUkraineCenterInput);
        subject.hasCrimeaUkraineCenter = (hasCrimeaUkraineCenterInput == "так");
        std::string hasDonbasUkraineCenterInput;
        std::getline(input, hasDonbasUkraineCenterInput);
        subject.hasDonbasUkraineCenter = (hasDonbasUkraineCenterInput == "так");
        std::string hasMilitaryDepartmentInput;
        std::getline(input, hasMilitaryDepartmentInput);
        subject.hasMilitaryDepartment = (hasMilitaryDepartmentInput == "так");
        input >> subject.foundationYear;
        input.ignore(std::numeric_limits<std::streamsize>::max(), '\n');

        registry.push_back(subject);
    }

    input.close();
}
else
{
    std::ofstream registry("register.data");
    registry.close();
}
}

void printRegistry(const std::vector<Subject>& registry, std::ostream& output)
{
    if(registry.size() != 0)
    {
        output << "\t\t\t\tВесь реєстр суб'єктів освітньої діяльності:" << std::endl;

        for (const auto& subject : registry) {

```

```

        output << "\nКод закладу в ЄДЕБО: " << subject.code << std::endl;
        output << "Повне найменування: " << subject.fullName << std::endl;
        output << "ЗАКЛАД ВИЩОЇ ОСВІТИ " << subject.shortName << std::endl;
        output << "Повне найменування (англ.): " << subject.fullNameEng <<
std::endl;

        output << "Форма власності: " << subject.ownershipForm << std::endl;
        output << "Найменування органу, до сфери управління якого належить заклад
освіти: " << subject.governingBody << std::endl;
        output << "Найменування посади, прізвище, ім'я, по батькові керівника
закладу освіти: " << subject.head << std::endl;
        output << "Місцезнаходження (юридична адреса): " << subject.location <<
std::endl;

        output << "Телефон / факс: " << subject.contact << std::endl;
        output << "Електронна пошта: " << subject.email << std::endl;
        output << "Веб-сайт: " << subject.website << std::endl;
        output << "ОЦ «Крим-Україна»: " << (subject.hasCrimeaUkraineCenter ? "так"
: "ні") << std::endl;
        output << "ОЦ «Донбас-Україна»: " << (subject.hasDonbasUkraineCenter ?
"так" : "ні") << std::endl;
        output << "Військова кафедра: " << (subject.hasMilitaryDepartment ? "так" :
"ні") << std::endl;
        output << "Рік заснування: " << subject.foundationYear << std::endl;
        output << std::endl;
    }

    std::cout << "_____ " << std::endl;
    std::cout << "\nОперація виконана успішно." << std::endl;
}
else
{
    std::cout << "В реєстрі ще не має записів." << std::endl;
}
}

void addRecord(std::vector<Subject>& registry)
{
    Subject subject;
    std::cout << "Код закладу в ЄДЕБО: ";
    std::cin >> subject.code;
    std::cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');

```

```
std::cout << "\nПовне найменування: ";
std::getline(std::cin, subject.fullName);

std::cout << "\nКоротка назва: ";
std::getline(std::cin, subject.shortName);

std::cout << "\nПовне найменування (англ.): ";
std::getline(std::cin, subject.fullNameEng);

std::cout << "\nФорма власності: ";
std::getline(std::cin, subject.ownershipForm);

std::cout << "\nНайменування органу, до сфери управління якого належить заклад
освіти: ";
std::getline(std::cin, subject.governingBody);

std::cout << "\nНайменування посади, прізвище, ім'я, по батькові керівника закладу
освіти: ";
std::getline(std::cin, subject.head);

std::cout << "\nМісцезнаходження (юридична адреса): ";
std::getline(std::cin, subject.location);

std::cout << "\nТелефон / факс: ";
std::getline(std::cin, subject.contact);

std::cout << "\nЕлектронна пошта: ";
std::getline(std::cin, subject.email);

std::cout << "\nВеб-сайт: ";
std::getline(std::cin, subject.website);

std::cout << "\nОЦ «Крим-Україна» (так/ні): ";
std::string hasCrimeaUkraineCenterInput;
std::getline(std::cin, hasCrimeaUkraineCenterInput);
subject.hasCrimeaUkraineCenter = (hasCrimeaUkraineCenterInput == "так");

std::cout << "\nОЦ «Донбас-Україна» (так/ні): ";
std::string hasDonbasUkraineCenterInput;
std::getline(std::cin, hasDonbasUkraineCenterInput);
subject.hasDonbasUkraineCenter = (hasDonbasUkraineCenterInput == "так");
```

```

std::cout << "\nВійськова кафедра (так/ні): ";
std::string hasMilitaryDepartmentInput;
std::getline(std::cin, hasMilitaryDepartmentInput);
subject.hasMilitaryDepartment = (hasMilitaryDepartmentInput == "так");

std::cout << "\nРік заснування: ";
std::cin >> subject.foundationYear;
std::cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');
std::cout << "\n_____ " << std::endl;

registry.push_back(subject);
}

void findSubject(std::vector<Subject>& registry, std::string code)
{
    bool found = false;

    for (int i = 0; i < registry.size(); i++)
    {
        if (registry[i].code == code)
        {
            std::cout << "\nКод закладу в ЄДЕБО: " << registry[i].code << std::endl;
            std::cout << "Повне найменування: " << registry[i].fullName << std::endl;
            std::cout << "Коротка назва: " << registry[i].shortName << std::endl;
            std::cout << "Повне найменування (англ.): " << registry[i].fullNameEng <<
std::endl;

            std::cout << "Форма власності: " << registry[i].ownershipForm << std::endl;
            std::cout << "Найменування органу, до сфери управління якого належить
заклад освіти: " << registry[i].governingBody << std::endl;
            std::cout << "Найменування посади, прізвище, ім'я, по батькові керівника
закладу освіти: " << registry[i].head << std::endl;
            std::cout << "Місцезнаходження (юридична адреса): " << registry[i].location
<< std::endl;

            std::cout << "Телефон / факс: " << registry[i].contact << std::endl;
            std::cout << "Електронна пошта: " << registry[i].email << std::endl;
            std::cout << "Веб-сайт: " << registry[i].website << std::endl;
            std::cout << "ОЦ «Крим-Україна»: " << (registry[i].hasCrimeaUkraineCenter ?
"так" : "ні") << std::endl;
            std::cout << "ОЦ «Донбас-Україна»: " << (registry[i].hasDonbasUkraineCenter
? "так" : "ні") << std::endl;

```

```

        std::cout << "Військова кафедра: " << (registry[i].hasMilitaryDepartment ?
"так" : "ні") << std::endl;
        std::cout << "Рік заснування: " << registry[i].foundationYear << std::endl;

        found = true;
    }
}

if(!found)
{
    std::cout << "_____ " << std::endl;
    std::cout << "\nЗапис з таким кодом закладу не знайдений у реєстрі." <<
std::endl;
}
}

void removeSubject(std::vector<Subject>& registry, std::string code)
{
    bool found = false;

    for (int i = 0; i < registry.size(); i++)
    {
        if (registry[i].code == code)
        {
            registry.erase(registry.begin() + i);

            found = true;
            std::cout << "_____ " << std::endl;
            std::cout << "\nЗапис був успішно знайдений та вилучений з реєстру." <<
std::endl;
        }
    }

    if(!found)
    {
        std::cout << "_____ " << std::endl;
        std::cout << "\nЗапис з таким кодом закладу не знайдений у реєстрі." <<
std::endl;
    }
}
}

```



```

void endWork(std::vector<Subject>& registry)
{
    std::ofstream output("register.data");

    if(output.is_open())
    {
        for(int i = 0; i < registry.size(); i++)
        {
            output << registry[i].code << std::endl;
            output << registry[i].fullName << std::endl;
            output << registry[i].shortName << std::endl;
            output << registry[i].fullNameEng << std::endl;
            output << registry[i].ownershipForm << std::endl;
            output << registry[i].governingBody << std::endl;
            output << registry[i].head << std::endl;
            output << registry[i].location << std::endl;
            output << registry[i].contact << std::endl;
            output << registry[i].email << std::endl;
            output << registry[i].website << std::endl;
            output << (registry[i].hasCrimeaUkraineCenter ? "так" : "ні") << std::endl;
            output << (registry[i].hasDonbasUkraineCenter ? "так" : "ні") << std::endl;
            output << (registry[i].hasMilitaryDepartment ? "так" : "ні") << std::endl;
            output << registry[i].foundationYear << std::endl;
        }

    }

    else
    {
        std::cout << "Не вдалося відкрити реєстр для завантаження даних." <<
std::endl;
    }

    output.close();
    std::cout << "Програма завершила роботу." << std::endl;

}

int main() {
    SetConsoleCP(1251); SetConsoleOutputCP(1251);

    std::vector<Subject> registry;

```

```

Subject subject;

registry_from_file(subject, registry);

int choice;
do
{
    std::cout << " _____" << std::endl;
    std::cout << "\nОберіть опцію:" << std::endl;
    std::cout << " _____" << std::endl;
    std::cout << "\n1. Вивести реєстр на екран" << std::endl;
    std::cout << "2. Записати реєстр у файл" << std::endl;
    std::cout << "3. Додати новий запис до реєстру" << std::endl;
    std::cout << "4. Пошук запису за кодом закладу" << std::endl;
    std::cout << "5. Вилучення запису з реєстру за кодом закладу" << std::endl;
    std::cout << "0. Завершити роботу програми" << std::endl;
    std::cout << " _____" << std::endl;
    std::cout << "\nВаш вибір: ";
    std::cin >> choice;
    std::cout << " _____" << std::endl <<
std::endl;
    std::cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');

    switch (choice)
    {
        case 1:
            printRegistry(registry, std::cout);
            break;
        case 2:
        {
            std::string file;

            std::cout << "Введіть назву файлу в який потрібно записати реєстр: ";
            std::cin >> file;
            std::cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');

            std::ofstream output(file);

            printRegistry(registry, output);

            output.close();

```

```

        break;
    }
    case 3:
        addRecord(registry);
        std::cout << "\nНовий запис був доданий до реєстру." << std::endl;
        break;
    case 4:
    {
        std::string code;

        std::cout << "Введіть код закладу в ЄДЕБО: ";
        std::cin >> code;

        findSubject(registry, code);
        std::cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');

        break;
    }
    case 5:
    {
        std::string code;

        std::cout << "Введіть код закладу в ЄДЕБО: ";
        std::cin >> code;
        std::cin.ignore(std::numeric_limits<std::streamsize>::max(), '\n');

        removeSubject(registry, code);
        break;
    }
    case 0:
        endWork(registry);
        break;
    default:
        std::cout << "Неправильний вибір. Будь ласка, спробуйте ще раз." <<
std::endl;
        break;
    }

    std::cout << "_____ " << std::endl;
    std::cout << "\nНатисніть будь-яку клавішу щоб продовжити далі!" <<
std::endl;;

```

```
        std::cin.get();

        std::system("cls");

    } while (choice != 0);

    return 0;
}
```