

Міністерство освіти і науки України  
Центральноукраїнський національний технічний університет  
Центр заочної та дистанційної освіти  
Кафедра кібербезпеки та програмного забезпечення

Звіт з лабораторної роботи № А[12]  
з дисципліни “Базові методології та технології програмування”  
на тему  
“Програмна реалізація абстрактних типів даних”

Виконав  
студент академічної групи  
КН-22з  
Михайленко С. Г.

Перевірив  
доцент  
Доренський О. П.

## Варіант 1

**Тема:** Програмна реалізація абстрактних типів даних

**Мета роботи** полягає у набутті ґрунтовних вмінь і практичних навичок об'єктного аналізу й проектування, створення класів С++ та тестування їх екземплярів, використання препроцесорних директив, макросів і макрооператорів під час реалізації програмних засобів у кросплатформовому середовищі Code::Blocks.

**Завдання:**

1. Як складову заголовкового файлу ModulesMykhailenko.h розробити клас ClassLab12\_Mykhailenko — формальне представлення абстракції сутності предметної області (об'єкта) за варіантом, — поведінка об'єкта якого реалізовує розв'язування задачі 7.1.
2. Реалізувати додаток Teacher, який видає 100 звукових сигналів і в текстовий файл TestResults.txt записує рядок "Встановлені вимоги порядку виконання лабораторної роботи порушено!", якщо файл проектуа main.cpp під час його компіляції знаходився не в \Lab12\prj, інакше — створює об'єкт класу ClassLab12\_Mykhailenko із заголовкового файлу ModulesMykhailenko.h та виконує його unit-тестування за тест-сьютом(ами) із \Lab12\TestSuite\, протоколюючи результати тестування в текстовий файл \Lab12\TestSuite\TestResults.txt.

Тест-сьют (у форматі TSV):

Edge	Volume	Seeds
5	125	100
10	1000	200

Лістинг коду:

ModulesMykhailenko.h:

```
#ifndef MODULESMYKHAILENKO_H_INCLUDED
#define MODULESMYKHAILENKO_H_INCLUDED

class ClassLab12_Mykhailenko {
private:
    float edge;
```

```
        unsigned int seeds;

public:
    ClassLab12_Mykhailenko(float, unsigned int);

    float getVolume();

    int getSeeds();

    bool setEdge(float);

    void setSeeds(unsigned int);

};
```

```
#endif // MODULESMYKHAILENKO_H_INCLUDED
```

### ModulesMykhailenko.cpp:

```
#include <cmath>
```

```
#include "ModulesMykhailenko.h"
```

```
ClassLab12_Mykhailenko::ClassLab12_Mykhailenko(float edgeToSet, unsigned int seedsToSet) {
```

```
    edge = edgeToSet;
```

```
    seeds = seedsToSet;
```

```
}
```

```
float ClassLab12_Mykhailenko::getVolume() {
```

```
    return pow(edge, 3);
```

```
}
```

```
int ClassLab12_Mykhailenko::getSeeds() {
```

```
    return seeds;
```

```
}
```

```
bool ClassLab12_Mykhailenko::setEdge(float edgeToSet) {
```

```
    if (edgeToSet > 0) {
```

```
        edge = edgeToSet;
```

```
        return true;
```

```

    } else {
        return false;
    }
}

void ClassLab12_Mykhailenko::setSeeds(unsigned int seedsToSet) {
    seeds = seedsToSet;
}

```

### Teacher.cpp:

```

#include <fstream>
#include <iostream>
#include <sstream>
#include <string>
#include <vector>
#include "ModulesMykhailenko.h"

using namespace std;

int main() {
    ofstream result;
    result.open("../TestSuite/TestResults.txt");
    if (!result)
        return 1;

    string path = __FILE__;

    if (path.find("\\lab12\\prj") == string::npos) {
        for (int i = 0; i < 100; i++)
            cout << "\a";

        result << "Встановлені вимоги порядку виконання лабораторної роботи порушено!" << endl;

        return 2;
    }
}

```

```

}

ifstream testsuite;
testsuite.open("../TestSuite/unit.txt");

vector<vector<string>> content;
vector<string> row;
string line, word;
unsigned int lines = 0;
bool firstLine = true;

if (testsuite.is_open()) {
    while(getline(testsuite, line)) {
        if (firstLine) {
            firstLine = false;
            continue;
        }

        row.clear();

        stringstream str(line);

        while(getline(str, word, '\u0009'))
            row.push_back(word);
        content.push_back(row);
        lines++;
    }
} else
    return 3;

float edge[lines];
float expectedVolume[lines];

```

```

unsigned int expectedSeeds[lines];

for (unsigned int i = 0; i < content.size(); i++) {
    edge[i] = stof(content[i][0]);
    expectedVolume[i] = stof(content[i][1]);
    expectedSeeds[i] = stoi(content[i][2]);
}

ClassLab12_Mykhailenko *classLab = new ClassLab12_Mykhailenko(edge[0],
expectedSeeds[0]);

for (unsigned int i = 0; i < lines; i++) {
    float volume = classLab->getVolume();
    unsigned int seeds = classLab->getSeeds();

    result << "Test " << i + 2 << ".1 " << (volume == expectedVolume[i] ?
"passed" : "failed") << endl;

    result << "Test " << i + 2 << ".2 " << (seeds == expectedSeeds[i] ?
"passed" : "failed") << endl;

    if (i + 1 != lines) {
        classLab->setEdge(edge[i + 1]);
        classLab->setSeeds(expectedSeeds[i + 1]);
    }
}
}

```

**Висновок:** під час виконання лабораторної роботи я навчився використовувати абстрактні типи даних.