

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

Звіт
Про виконання Лабораторії роботи №10
з дисципліни
“Базові методології та технології програмування”
на тему
«Реалізація програмних модулів оброблення даних складових типів з
файловим введенням/виведенням»

Виконав
Студент академічної групи КН-22
Гончарук О.С.

Перевірив
викладач
Собінов О.Г

Мета роботи: Набуття ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою програмування C++ програмних модулів створення й оброблення даних типів масив, структура, об'єднання, множина, перелік, перетворення типів даних, використання файлових потоків та функцій стандартних бібліотек для оброблення символічної інформації.

Завдання до лабораторної роботи:

1. Реалізувати програмні модулі розв'язування задач 10.1-10.3 як складові статичної бібліотеки libModulesGoncharuk.a (проект ModulesGoncharuk лабораторних робіт № 8-9).
2. Реалізувати тестовий драйвер автоматизованої перевірки програмних модулів розв'язування задач 10.1-10.3.

Варіант 3

<https://github.com/odorenskyi/>

ВАРІАНТ 3

— ВХІДНИЙ ТЕКСТ - ВМІСТ ВХІДНОГО ТЕКСТОВОГО ФАЙЛУ —

Довільний текст українською мовою.

— ЗАДАЧА 10.1 —

У вихідний текстовий файл записати:

- авторську інформацію: ім'я й прізвище розробника модуля, установа/організація, місто, країна, рік розробки;
- кількість абзаців у тексті із вхідного файлу;
- повідомлення, чи є у тексті з вхідного файлу слова “Україна”, “університет” та “блокнот” (у програмі слід реалізувати розрізнення слів, наприклад, “університет” та “університету” тощо).

— ЗАДАЧА 10.2 —

У вхідний текстовий файл дописати:

- кількість знаків пунктуації у ньому, дату й час дозапису інформації.

— ЗАДАЧА 10.3 —

Вхідні дані – числові значення x , y , z та натуральне число b . У вихідний текстовий файл дописати:

- результати виконання функцій із заголовкового файлу Modules/Прізвище.h `s_calculation` з аргументами x , y , z ;
- число b у двійковому коді.



- Мова повідомлень – українська (наприклад, якщо у вихідний файл записується кількість символів у вхідному файлі, то модуль повинен сформувати й записати/дописати повноцінне речення: “У файлі ВхФайл.txt міститься 257 символів.”).
- Вхідний файл *.txt створюється користувачем, у який за допомогою текстового редактора (у Windows – Блокнот) записується вхідний текст відповідно до завдання; вихідний файл створюється програмним модулем; імена вхідного й вихідного файлів є параметрами відповідного модуля.
- Перед читання/записом з/у файлового потіку слід реалізувати перевірку його відкриття; після завершення – закрити всі відкриті файлові потоки.
- Оброблення текстових файлів рекомендовано реалізувати за допомогою файлових потоків `ofstream` та `ifstream` <fstream> C++.
- Для отримання локальної дати й часу ОС можна використати стандартні функції `time`, `ctime`, `localtime`, `asctime`, реалізовані у `ctime / time.h`.

Лістинг коду:

ModulesGoncharuk.cpp:

```
#include <iostream>
```

```
#include <locale>
```

```
#include <ctime>
```

```
#include <cmath>
```

```
#include <limits>
```

```
#include <string>
```

```
#include <fstream>
```

```
#include <bitset>
```

```
#include <codecvt>
```

```
#include "ModulesGoncharuk.h"
```

```
using namespace std;
```

```
void SoftwareDeveloper()
```

```
void s_calculation(char choice);
```

```
{
```

```
    system("chcp 1251");
```

```
    system("cls");
```

```
    setlocale(LC_ALL, "ukr");
```

```
    cout << "\n ----- \n"
```

```
        << " | Oleksandr Goncharuk |"
```

```
        << " | Олександр Гончарук | "
```

```
        << "\n ----- © All Rights Reserved ----- \n \n \n \n";
```

```
}
```

```
using namespace std;
```

```
wstring characters = L",.!?:;)\\" (Љ"№%*_ -+=\\0123456789абвггдесжзіїйклмнопрстуфхцшщьюя";
```

```
float s_calculation(int x, int y, int z) {
    return (1.0/2.0)*pow(fabs(2*z-pow(x,2))/sin(x),3)/sqrt(1+fabs(cos(x))+2*3.14*y);
}
```

```
Deposit getPayment(float value, int months) {
    Deposit deposit;
    if (months == 6 || months == 12) {
        float interest;
        deposit.totalInterest = months == 12 ? 13 : 11/(float)2;
        interest = deposit.totalInterest / 100 / months;
        deposit.monthlyInterestPaid = round(value*interest*100)/100;
    } else {
        deposit.totalInterest = -1;
        deposit.monthlyInterestPaid = -1;
    }
    return deposit;
}
```

```
Size getSize(int slovakSize) {
    Size size;
    if (slovakSize >= 6 && slovakSize <= 10) {
        size.french = slovakSize - 4;
        switch (slovakSize)
        {
            case 6:
                size.international = "S";
                break;
            case 7:
                size.international = "M";
                break;
```

```

        case 8:

            size.international = "L";

            break;

        case 9:

            size.international = "XL";

            break;

        case 10:

            size.international = "XXL";

            break;

        default:

            size.international = "Error";

    }

} else {

    size.french = -1;

    size.international = "Error";

}

return size;

}

```

```

int t9_3(int number) {

    unsigned int count = 0;

    if (number > 0 || number <= 7483650) {

        bool set = number & 1;

        while (number) {

            count += (number & 1) == set;

            number >>= 1;

        }

    } else {

        return -1;

    }

    return count;
}

```

```
}
```

```
bool validateCharacter(wchar_t character) {  
    for (int i = 0; i < characters.length(); i++)  
        if (character == characters.at(i))  
            return true;  
  
    return false;  
}
```

```
int t10_1(string inputFile, string outputFile) {  
    wstring line;  
  
    wstring words[4] = {L"програма", L"модуль", L"студент", L"програміст"};  
  
    bool found = false;  
  
    int number = 0;  
  
    wifstream indata;  
  
    indata.open(inputFile);  
  
    ofstream outdata;  
  
    outdata.open(outputFile);  
  
  
    if (!indata || !outdata)  
        return 1;  
  
  
    indata.imbue(locale(locale(), new codecvt_utf8<wchar_t>));  
  
  
    while (getline(indata, line)) {  
        number += line.length();  
  
  
        for (int i = 0; i < line.length(); i++) {  
            line[i] = towlower(line[i]);  
  
            if (!validateCharacter(line[i]))  
                return 2;  
        }  
    }  
}
```

```

    }

    if (!found) {
        for (int i = 0; i < 4; i++) {
            int index = line.find(words[i]);

            if (index != wstring::npos && line.length()-words[i].length()-index == 0) { // слово у кінці

                found = true;

                goto out_found;
            }

            for (int j = 0; j < 9; j++) { // крапка, кома, знак оклику, знак питання, двокрапка, крапка з
комою, закриваюча дужка, лапки, пробіл

                wstring wordToFind = words[i] + characters.at(j);

                if (line.find(wordToFind) != wstring::npos) {

                    found = true;

                    goto out_found;
                }
            }
        }
    }

    out_found: continue;
}

    outdata << "Розробник: Гончарук Олександр" << endl << "Установа/організація: Центральноукраїнський
національний технічний університет" << endl << "Місто: Кропивницький" << endl << "Країна: Україна" << endl <<
"Рік розробки: 2023" << endl << endl;

    outdata << "Кількість символів у файлі: " << number << endl;

    outdata << "У вхідному файлі " << ((found) ? "наявні" : "відсутні") << " слова \"Україна\", \"університет\",
\"блокнот\".\"\".\" << endl;

    return 0;

```

```
}
```

```
int t10_2(string file) {
```

```
    wchar_t character;
```

```
    time_t rawtime;
```

```
    time(&rawtime);
```

```
    int number = 0;
```

```
    wifstream indata;
```

```
    indata.open(file);
```

```
    ofstream outdata;
```

```
    outdata.open(file, ios_base::app);
```

```
    if (!indata || !outdata)
```

```
        return 1;
```

```
    indata.imbue(locale(locale(), new codecvt_utf8<wchar_t>));
```

```
    while (indata >> character) {
```

```
        if (!validateCharacter(tolower(character)))
```

```
            return 2;
```

```
        for (int i = 0; i < 10; i++)
```

```
            if (character == characters[i + 22]) // 0-9
```

```
                number += 1;
```

```
    }
```

```
    outdata << endl << "Кількість цифр у файлі: " << number << endl;
```

```
    outdata << "Дата й час дозапису: " << ctime(&rawtime);
```

```
    return 0;
```

```
}
```



```
int t10_3(string file, int x, int y, int z, int b) {  
  
    wchar_t character;  
  
    time_t rawtime;  
  
    time(&rawtime);  
  
    int number = 0;  
  
  
    ofstream data;  
  
    data.open(file, ios_base::app);  
  
  
    if (!data)  
        return 1;  
  
  
    data << endl << "s = " << s_calculation(x, y, z) << endl;  
  
    if (b > 0) {  
  
        string binary = bitset<numeric_limits<int>::digits>(b).to_string();  
  
        binary.erase(0, binary.find_first_not_of('0'));  
  
        data << "b у двійковому коді: " << binary << endl;  
  
    }  
  
    else  
  
        data << "b — не натуральне число" << endl;  
  
  
    return 0;  
  
}
```

ЛІСТИНГ коду:

ModulGoncharuk.h:

```
#ifndef MODULGONCHARUK_H_INCLUDED
#define MODULGONCHARUK_H_INCLUDED

#include <string>

void SoftwareDeveloper(void);

using std::string;

float s_calculation(int x, int y, int z);

typedef struct Deposit {
    float totalInterest;
    float monthlyInterestPaid;
} Deposit;

typedef struct Size {
    int french;
    string international;
} Size;

Deposit getPayment(float value, int months);

Size getSize(int slovakSize);

int t9_3(int number);

bool validateCharacter(wchar_t character);

int t10_1(string inputFile, string outputFile);
```

```
int t10_2(string file);
```

```
int t10_3(string file, int x, int y, int z, int b);
```

```
#endif // MODULGONCHARUK_H_INCLUDED
```

Лістинг коду:

TestDriver.cpp:

```
#include "ModulesGoncharuk.h"
```

```
#include <iostream>
```

```
#include <locale>
```

```
#include <fstream>
```

```
#include <codecvt>
```

```
#include <string>
```

```
using namespace std;
```

```
string input = "input.txt";
```

```
string output = "output.txt";
```

```
bool createInput(wstring content) {
```

```
    wofstream data;
```

```
    data.open(input);
```

```
    data.imbue(locale(locale(), new codecvt_utf8<wchar_t>));
```

```
    if (!data) {
```

```
        cout << "Помилка запису вхідних даних." << endl;
```

```
        return false;
```

```
    }
```

```
    data << content << endl;
```

```
    return true;
```

```

}

bool test1_1() {

    wstring line;

    const wstring lines[3] = {L"Розробник: Гончарук Олександр", L"Установа/організація: Центральноукраїнський національний технічний університет", L"Рік розробки: 2023"};

    bool linesFound[3] = {false, false, false};

    int currentLine = 0;

    if(!createInput(L"тест"))

        return false;


    t10_1(input, output);

    wifstream indata;

    indata.open(output);

    indata.imbue(locale(locale(), new codecvt_utf8<wchar_t>));

    while (getline(indata,line)) {

        if (line.find(lines[currentLine]) != wstring::npos) {

            linesFound[currentLine] = true;

            currentLine++;

        }

        if (linesFound[0] && linesFound[1] && linesFound[2])

            return true;

    }

    return false;

}

bool test1(wstring data, bool wordPresent) {

    wstring line;


    if(!createInput(data))

        return false;

```

```

t10_1(input, output);

wifstream indata;

indata.open(output);

indata.imbue(locale(locale(), new codecvt_utf8<wchar_t>));


while (getline(indata,line)) {

    if ((wordPresent && line.find(L"наявні") != wstring::npos) || (!wordPresent && line.find(L"відсутні") !=
wstring::npos))

        return true;

}


return false;

}

bool test2_1() {

    wstring line;

    if(!createInput(L"текст"))

        return false;

    t10_2(input);

    wifstream indata;

    indata.open(input);

    indata.imbue(locale(locale(), new codecvt_utf8<wchar_t>));

    while (getline(indata,line)) {

        if (line.find(L"Дата й час дозапису: ") != wstring::npos)

            return true;

    }


    return false;

}

bool test2(wstring data, int digits) {

    wstring line;

    if(!createInput(data))

        return false;

```

```

t10_2(input);

wifstream indata;

indata.open(input);

indata.imbue(locale(locale(), new codecvt_utf8<wchar_t>));

while (getline(indata,line)) {

    if (line.find(L"Кількість цифр у файлі: " + digits) != wstring::npos)

        return true;

}

return false;

}

bool test3(int x, int y, int z, int b, wstring s, wstring bin) {

    wstring line;

    bool firstValid = false;

    t10_3(output, x, y, z, b);

    wifstream indata;

    indata.open(output);

    indata.imbue(locale(locale(), new codecvt_utf8<wchar_t>));

    while (getline(indata,line)) {

        if (!firstValid && (line.find(L"s = " + s) != wstring::npos))

            firstValid = true;

        else if (firstValid && (line.find(L"b у двійковому коді: " + bin) != wstring::npos))

            return true;

    }

    return false;

}

int main() {

    const wstring data1_1[] = {L"Університет", L"Університету", L"Україна", L"України"};

    const bool wordPresent[] = {true, false, true, false};

    const wstring data2_1[] = {L"Блокнот", L"0123456789", L"12", L"00000"};

```

```

const int digits[] = {0, 10, 2, 5};

const int x[] = {7, 2, 9, 0, 7};

const int y[] = {2, 45, 0, 0, 5};

const int z[] = {1, 6, 1, 0, 4};

const int b[] = {1, 3, 8, 100, 127};

const wstring s[] = {L"7.84054", L"268.187", L"3.84147", L"1", L"35775.4"};

const wstring bin[] = {L"1", L"11", L"1000", L"1100100", L"1111111"};

setlocale(LC_ALL, "");

for (int i = 0; i < 3; i++) {

    for (int j = 0; j < 5; j++) {

        bool value = false;

        wstring description;

        if (i < 2 && j == 0) {

            description = L"виведення повідомлення";

            if (i == 0)

                value = test1_1();

            else if (i == 1)

                value = test2_1();

        }

        else if (i == 0) {

            value = test1(data1_1[j-1], wordPresent[j-1]);

            description = L"вхідні дані: " + data1_1[j-1] + L", очікуваний результат: слова " + (wordPresent[j-1] ? L"наявні"
: L"відсутні");

        }

        else if (i == 1) {

            value = test2(data2_1[j-1], digits[j-1]);

            string tmpstr = to_string(digits[j-1]);

            wstring wstr(tmpstr.begin(), tmpstr.end());

            description = L"вхідні дані: " + data2_1[j-1] + L", очікуваний результат: кількість чисел: " +

wstr;

```

```

    }

    else if (i == 2) {

        value = test3(x[j], y[j], z[j], b[j], s[j], bin[j]);

        string xtmpstr = to_string(digits[j-1]);

        wstring xwstr(xtmpstr.begin(), xtmpstr.end());

        string ytmpstr = to_string(digits[j-1]);

        wstring ywstr(ytmpstr.begin(), ytmpstr.end());

        string ztmpstr = to_string(digits[j-1]);

        wstring zwstr(ztmpstr.begin(), ztmpstr.end());

        string btmpstr = to_string(digits[j-1]);

        wstring bwstr(btmpstr.begin(), btmpstr.end());

        description = L"вхідні дані: x = " + xwstr + L", y = " + ywstr + L", z = " + zwstr + L", b = " +
        bwstr + L", очікуваний результат: s = " + s[j] + L", b у двійковому коді: " + bin[j];

    }

    wcout << "Test " << i + 1 << "." << j + 1 << " (" << description << ") " << (value ? "passed" : "failed") <<
endl;

}

}

return 0;

}

```


TestSuite10.1

Назва тестового набору	Задача 10.1
Назва проекту / ПЗ	Лабораторна робота 10
Рівень тестування	модульний / Unit Testing
Автор тест-сююта	Гончарук Олександр Святославович
Виконавець	Гончарук Олександр Святославович

Ід-р тест-кейса	Дії (кроки)	Очікуваний результат	Результат тестування
1.1	Довільні правильні вхідні дані	У вихідному файлі є повідомлення: Розробник: Гончарук Олександр Установа/організація: Центральноукраїнський національний технічний університет Рік розробки: 2023	passed
1.2	Україна	У вхідному файлі наявні слова «Україна», «університет», «блокнот»	passed
1.3	університет	У вхідному файлі відсутні слова «Україна», «університет», «блокнот»	passed

1.4	блокнот	У вхідному файлі наявні слова «Україна», «університет», «блокнот»	passed
-----	---------	---	--------

TestSuite10.2

Ід-р тест-кейса	Дії (кроки)	Очікуваний результат	Результат тестування
2.1	Довільні правильні вхідні дані	<p>У вхідному файлі є повідомлення:</p> <p>Дата й час дозапису: ДТЯ НМ Ч ГГ:ХХ:СС РРРР</p> <p>де:</p> <p>ДТЯ — день тижня</p> <p>НМ — назва місяця</p> <p>Ч — число</p> <p>ГГ:ХХ:СС — години, хвилини, секунди відповідно</p> <p>РРРР — рік</p> <p>Дата і час мають відповідати тим, що встановлені на комп'ютері.</p>	passed
2.2	Україна	Кількість пробілів у файлі: 0	passed
2.3	9876543210	Кількість пробілів у файлі: 0	passed
2.4	Україна незалежна держава	Кількість пробілів у файлі: 2	passed

2.5	блокнот чудовий	Кількість пробілів у файлі: 1	passed
-----	-----------------	-------------------------------	--------

TestSuite10.3

Ід-р тест- кейса	Дії (кроки)	Очікуваний результат	Результат тестування
3.1	$x = 9, y = 88, z = 7, b = 6$	$s = 91234.1$ b у двійковому коді: 110	passed
3.2	$x = 4, y = 64, z = 4, b = 432$	$s = -28.9518$ b у двійковому коді: 110110000	passed

Висновок

Завантажений власний Git-репозиторій <https://github.com/odorenskyi/Honcharuk-Oleksandr-KN22>. У \Lab10 заповнено файл README.md, створено теки prj, Software, TestSuite, Report. До звіту включено тему мету завдання. Здійснено аналіз і постановку задач 10.1, 10.2, 10.3. Виконано аналіз вимог, проектування архітектури, детальне проектування програмних модулів розв'язування задач 10.1, 10.2, 10.3. Розроблено набори контрольних прикладів до задач 10.1, 10.2, 10.3 задля виконання модульного тестування (Unit testing) модулів C++. В Code::Blocks IDE відкрито проект статичної бібліотеки ModulGoncharuk з \Lab8\prj, створений під час виконання лабораторної роботи №8. За отриманими під час проектування програмних модулів артефактами виконано конструювання функцій: мовою програмування C++ реалізовано функції, які за наданим інтерфейсом реалізують розв'язування задач 10.1, 10.2 та 10.3 відповідно. Скомпільовано ModulesGoncharuk. Відкрито проект заголовкового файлу ModulGoncharuk із \Lab8\prj, створений під час виконання лабораторної роботи №8, та доповнено його прототипами функцій розв'язування задач 10.1–10.3. В Code::Blocks IDE у \prj створено проект консольного додатка C++, під назвою TestDriver. Реалізовано тестовий драйвер для виконання розроблених тестових наборів до задач 10.1-10.3 і за його допомогою виконано модульне тестування функцій зі статичної бібліотеки libModulesGoncharuk.a. Результати unit-тестування задокументовано.

В ході роботи над лабораторною роботою набув ґрунтовних вмінь і практичних навичок реалізації технології модульного програмування, застосування операторів C/C++ арифметичних, логічних, побітових операцій, умови, циклів та вибору під час розроблення статичних бібліотек, заголовкових файлів та програмних засобів у кросплатформовому середовищі Code::Blocks.