

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет
Механіко-технологічний факультет

ЗВІТ

Про виконання лабораторної роботи № 10

З навчальної дисципліни: “Базові методології та технології
програмування”

На тему: “Реалізація програмних модулів оброблення даних складових
типів з файловим I/O”.

ВИКОНАВ

студент академічної групи

КБ-22-1

Паращенко Д.С

ПЕРЕВІРИВ

Викладач

Собінов.О.Г

Кропивницьк - 2022

ТЕМА: Реалізація програмних модулів оброблення даних складових типів з файловим I/O.

МЕТА: полягає у набутті ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою 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:

ВАРІАНТ 9

— ВХІДНИЙ ТЕКСТ - ВМІСТ ВХІДНОГО ТЕКСТОВОГО ФАЙЛУ —

Довільний текст з довільної кількості абзаців українською або англійською.

— ЗАДАЧА 10.1 —

У вихідний текстовий файл записати:

- авторську інформацію: ім'я й прізвище розробника модуля, установа/організація, місто, країна, рік розробки;
- текст із вхідного файлу з переставленими місцями абзацами (перший – останній, другий – передостанній і т.д.);
- повідомлення, до англійської чи української мови належить текст у вхідному файлі.

— ЗАДАЧА 10.2 —

У вихідний текстовий файл дописати:

- кількість ком та крапок 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 `ModulesПрізвище.hs_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 сформулювати й записати/дописати повноцінне речення: “У файлі `ВхФайл.txt` міститься 257 символів.”).
- Вхідний файл `*.txt` створиться 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 файловий потік 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`.

Задача 10.1

```
#include <iostream>
#include <fstream>
#include <cstring>
#include <ctime>
#include <vector>
#include <algorithm>
#include <Windows.h>
```

```
using namespace std;
```

```
int main() {
    SetConsoleOutputCP(1251);
    SetConsoleCP(1251);
    SetConsoleOutputCP(CP_UTF8);
    // Відкриваємо вхідний файл
    ifstream inputFile("input.txt");
    if (!inputFile) {
        cerr << "Не вдалося відкрити вхідний файл" << endl;
        return 1;
    }

    // Зчитуємо текст з вхідного файлу
    string inputText;
    getline(inputFile, inputText, '\0');

    // Визначаємо, до якої мови належить текст
    bool isUkrainian = false;
```

```
for (char c : inputText) {  
    if ((c >= 0x0400 && c <= 0x04FF) || (c >= 0x0500 && c <= 0x052F) || (c ==  
0x0456) || (c == 0x0491) || (c == 0x0490) || (c == 0x0407) || (c == 0x0406)) {  
        isUkrainian = true;  
        break;  
    }  
}
```

```
// Закриваємо вхідний файл
```

```
inputFile.close();
```

```
// Відкриваємо вихідний файл
```

```
ofstream outputFile("output.txt", ios::out | ios::binary);
```

```
if (!outputFile) {
```

```
    cerr << "Не вдалося відкрити вихідний файл" << endl;
```

```
    return 1;
```

```
}
```

```
// Додаємо маркер порядку байтів UTF-8 до вихідного файлу
```

```
outputFile << char(0xEF) << char(0xBB) << char(0xBF);
```

```
// Змінюємо порядок абзаців у тексті
```

```
vector<string> paragraphs;
```

```
size_t start = 0;
```

```
size_t end = inputText.find("\n\n");
```

```
while (end != string::npos) {
```

```
    paragraphs.push_back(inputText.substr(start, end - start));
```

```
    start = end + 2;
```

```
    end = inputText.find("\n\n", start);
```

```

    }
    paragraphs.push_back(inputText.substr(start));
    reverse(paragraphs.begin(), paragraphs.end());

    // Виводимо змінений текст у консоль
    for (const string& paragraph : paragraphs) {
        cout << paragraph << endl << endl;
    }

    // Записуємо змінений текст у вихідний файл
    for (const string& paragraph : paragraphs) {
        outputFile << paragraph << endl << endl;
    }

    // Записуємо інформацію про мову тексту у вихідний файл
    if (isUkrainian) {
        outputFile << "Мова тексту: Українська\n";
    } else {
        outputFile << "Мова тексту: Інша\n";
    }

    // Виводимо рік
    time_t now = time(0);
    tm* ltm = localtime(&now);
    int year = 1900 + ltm->tm_year;
    cout << "Рік розробки: " << year << endl;

    // Закриваємо вихідний файл

```

```
outputFile.close();
```

```
return 0;
```

```
}
```

Задача 10.2

```
#include <iostream>
```

```
#include <fstream>
```

```
#include <cstring>
```

```
#include <ctime>
```

```
#include <vector>
```

```
#include <algorithm>
```

```
#include <Windows.h>
```

```
using namespace std;
```

```
int main() {
```

```
    SetConsoleOutputCP(1251);
```

```
    SetConsoleCP(1251);
```

```
    SetConsoleOutputCP(CP_UTF8);
```

```
    // Відкриваємо вхідний файл
```

```
    ifstream inputFile("input.txt");
```

```
    if (!inputFile) {
```

```
        cerr << "Не вдалося відкрити вхідний файл" << endl;
```

```
        return 1;
```

```
    }
```

```
    // Зчитуємо текст з вхідного файлу
```

```
string inputText;
getline(inputFile, inputText, '\0');

// Визначаємо, до якої мови належить текст
bool isUkrainian = false;
for (char c : inputText) {
    if ((c >= 0x0400 && c <= 0x04FF) || (c >= 0x0500 && c <= 0x052F) || (c ==
0x0456) || (c == 0x0491) || (c == 0x0490) || (c == 0x0407) || (c == 0x0406)) {
        isUkrainian = true;
        break;
    }
}

// Закриваємо вхідний файл
inputFile.close();

// Відкриваємо вихідний файл
ofstream outputFile("output.txt", ios::out | ios::binary);
if (!outputFile) {
    cerr << "Не вдалося відкрити вихідний файл" << endl;
    return 1;
}

// Додаємо маркер порядку байтів UTF-8 до вихідного файлу
outputFile << char(0xEF) << char(0xBB) << char(0xBF);

// Підраховуємо кількість ком та крапок у тексті
int numCommas = 0;
int numPeriods = 0;
for (char c : inputText) {
```



```
if (c == ',') {
    numCommas++;
} else if (c == '.') {
    numPeriods++;
}
}

// Виводимо кількість ком та крапок у консоль
cout << "Кількість ком: " << numCommas << endl;
cout << "Кількість крапок: " << numPeriods << endl;

// Записуємо кількість ком та крапок у вихідний файл
outputFile << "Кількість ком: " << numCommas << endl;
outputFile << "Кількість крапок: " << numPeriods << endl;

// Виводимо дату та час дозапису у консоль
time_t currentTime = time(nullptr);
tm* localTime = localtime(&currentTime);
cout << "Дата та час дозапису: " << asctime(localTime) << endl;

// Записуємо дату та час дозапису у вихідний файл
outputFile << "Дата та час дозапису: " << asctime(localTime) << endl;

// Закриваємо вихідний файл
outputFile.close();

return 0;
}
```

Задача 10.3

```
#include <iostream>
#include <fstream>
#include <cstring>
#include <ctime>
#include <vector>
#include <algorithm>
#include <bitset>
#include <Windows.h>
#include "ModulesParashchenko.h"
```

```
using namespace std;
```

```
int main() {
    SetConsoleOutputCP(1251);
    SetConsoleCP(1251);
    SetConsoleOutputCP(CP_UTF8);

    // Отримуємо значення x, z та b від користувача
    double x, z;
    int b;
    cout << "Введіть значення x: ";
    cin >> x;
    cout << "Введіть значення z: ";
    cin >> z;
    cout << "Введіть натуральне число b: ";
    cin >> b;
```

```
// Відкриваємо вхідний файл
ifstream inputFile("input.txt");
if (!inputFile) {
    cerr << "Не вдалося відкрити вхідний файл" << endl;
    return 1;
}

// Зчитуємо текст з вхідного файлу
string inputText;
getline(inputFile, inputText, '\0');

// Визначаємо, до якої мови належить текст
bool isUkrainian = false;
for (char c : inputText) {
    if ((c >= 0x0400 && c <= 0x04FF) || (c >= 0x0500 && c <= 0x052F) || (c
== 0x0456) || (c == 0x0491) || (c == 0x0490) || (c == 0x040...
        if (c >= 'a' && c <= 'я') {
            isUkrainian = true;
            break;
        }
    }
}

// Відкриваємо вихідний файл для дописування
ofstream outputFile("output.txt", ios_base::app);
if (!outputFile) {
    cerr << "Не вдалося відкрити вихідний файл" << endl;
    return 1;
}
```

```
// Розбиваємо текст на абзаци
vector<string> paragraphs;
string paragraph;
for (char c : inputText) {
    if (c == '\n' && !paragraph.empty()) {
        paragraphs.push_back(paragraph);
        paragraph.clear();
    } else {
        paragraph += c;
    }
}
if (!paragraph.empty()) {
    paragraphs.push_back(paragraph);
}

// Визначаємо кодування файлу
string encoding;
if (isUkrainian) {
    encoding = "UTF-8";
} else {
    encoding = "ASCII";
}

// Виводимо результати
time_t now = time(nullptr);
outputFile << "Результати розрахунків від " << ctime(&now);
outputFile << "Прізвище та ім'я автора: " << author_name() << endl;

double x = 3.14;
```

```

double z = 2.71;

double S = s_calculation(x, z);

outputFile << "Результати виконання функції s_calculation з аргументами x
= " << x << " та z = " << z << ": " << S << endl;


int b = 10;

string binaryB;

while (b != 0) {
    binaryB = to_string(b % 2) + binaryB;
    b /= 2;
}

outputFile << "Число " << b << " у двійковому коді: " << binaryB << endl;

outputFile.close();

return 0;
}

```

Код

```

#include <iostream>
#include <fstream>
#include <cstring>
#include <ctime>
#include <vector>
#include <algorithm>
#include <Windows.h>
#include "s_calculation.h"
#include "ModulesParashchenko.h"

```

```
#include <bitset>

using namespace std;

int main() {
    //SetConsoleOutputCP(1251);
    //SetConsoleCP(1251);
    SetConsoleOutputCP(CP_UTF8);
    // Відкриваємо вхідний файл
    ifstream inputFile("input.txt");
    if (!inputFile) {
        cerr << "Не вдалося відкрити вхідний файл" << endl;
        return 1;
    }

    // Зчитуємо текст з вхідного файлу
    string inputText;
    getline(inputFile, inputText, '\0');

    // Визначаємо, до якої мови належить текст
    bool isUkrainian = false;
    for (char c : inputText) {
        if ((c >= 0x0400 && c <= 0x04FF) || (c >= 0x0500 && c <= 0x052F) || (c ==
0x0456) || (c == 0x0491) || (c == 0x0490) || (c == 0x0407) || (c == 0x0406)) {
            isUkrainian = true;
            break;
        }
    }
}
```

```
// Закриваємо вхідний файл
inputFile.close();

// Відкриваємо вихідний файл
ofstream outputFile("output.txt", ios::out | ios::binary);
if (!outputFile) {
    cerr << "Не вдалося відкрити вихідний файл" << endl;
    return 1;
}

// Додаємо маркер порядку байтів UTF-8 до вихідного файлу
outputFile << char(0xEF) << char(0xBB) << char(0xBF);

// Підраховуємо кількість ком та крапок у тексті
int numCommas = 0;
int numPeriods = 0;
for (char c : inputText) {
    if (c == ',') {
        numCommas++;
    } else if (c == '.') {
        numPeriods++;
    }
}

// Виводимо кількість ком та крапок у консоль
cout << "Кількість ком: " << numCommas << endl;
cout << "Кількість крапок: " << numPeriods << endl;

// Записуємо кількість ком та крапок у вихідний файл
```

```

outputFile << "Кількість ком: " << numCommas << endl;
outputFile << "Кількість крапок: " << numPeriods << endl;

// Виводимо дату та час дозапису у консоль
time_t currentTime = time(nullptr);
tm* localTime = localtime(&currentTime);
cout << "Дата та час дозапису: " << asctime(localTime) << endl;

// Записуємо дату та час дозапису у вихідний файл
outputFile << "Дата та час дозапису: " << asctime(localTime) << endl;

// Змінюємо порядок абзаців у тексті
vector<string> paragraphs;
size_t start = 0;
size_t end = inputText.find("\n\n");
while (end != string::npos) {
    paragraphs.push_back(inputText.substr(start, end - start));
    start = end + 2;
    end = inputText.find("\n\n", start);
}
paragraphs.push_back(inputText.substr(start));
reverse(paragraphs.begin(), paragraphs.end());

// Виводимо змінений текст у консоль
for (const string& paragraph : paragraphs) {
    cout << paragraph << endl << endl;
}

```



```

// Записуємо змінений текст у вихідний файл
for (const string& paragraph : paragraphs) {
    outputFile << paragraph << endl << endl;
}

// Записуємо інформацію про мову тексту у вихідний файл
if (isUkrainian) {
    outputFile << "Мова тексту: Українська\n";
} else {
    outputFile << "Мова тексту: Інша\n";
}

// Виводимо рік
time_t now = time(0);
tm* ltm = localtime(&now);
int year = 1900 + ltm->tm_year;
cout << "Рік розробки: " << year << endl;

// Зчитуємо вхідні дані
double x = 2, z = 50;
int b;
inputFile >> x >> z >> b;

// Обчислюємо результат виконання функції s_calculation з аргументами
x, z
double result = s_calculation(x, z);

// Записуємо результат вихідної функції у вихідний файл

```

```
outputFile << "Результат виконання функції s_calculation з аргументами  
x = " << x << ", z = " << z << " дорівнює " << result << endl;
```

```
// Конвертуємо число b в двійковий формат та записуємо його у  
вихідний файл
```

```
string binary = bitset<32>(b).to_string();
```

```
outputFile << "Число " << b << " у двійковій формі: " << binary << endl;
```

```
// Закриваємо вихідний файл
```

```
outputFile.close();
```

```
return 0;
```

```
}
```

Назва тестового набору Test Suite Description	BMTP_LW10_TS1
Назва проекту / ПЗ Name of Project / Software	main.exe
Рівень тестування Level of Testing	Модульний
Автор тест-сюита Test Suite Author	Паращенко Д. С.
Виконавець Implementer	Паращенко Д. С.

Ід-р тест - кей са / Test Cas e ID	Дії (кроки) / Action (Test Steps)	Очікуваний результат / Expected Result	Результат тестуванн я (пройшов /не вдало ся/ заблокова ний) / Test Result (passed/fa iled/ blocked)
T_1	Вводим: ,.	<p>Кількість ком: 1</p> <p>Кількість крапок: 1</p> <p>Дата та час дозапису: Thu May 04 04:14:30 2023</p> <p>„</p> <p>Рік розробки: 2023</p> <p>Process returned 0 (0x0) execution time : 0.110 s</p> <p>Press any key to continue.</p>	passed
T_2	Вводим: Привіт світ.,,,	<p>Кількість ком: 0</p> <p>Кількість крапок: 4</p> <p>Дата та час дозапису: Thu May 04 04:13:57 2023</p> <p>Прізвище Ім'я: Паращенко Денис.</p>	passed

		<p>Установа/організація: Центральноукраїнський національний технічний університет.</p> <p>Місто: Кропивницьк.</p> <p>Країна: Україна.</p> <p>Рік розробки: 2023</p> <p>Process returned 0 (0x0) execution time : 0.078 s</p> <p>Press any key to continue.</p>	
T_3	Вводим: Hello world.	<p>Кількість ком: 0</p> <p>Кількість крапок: 1</p> <p>Дата та час дозапису: Thu May 04 04:16:35 2023</p> <p>Hello world.</p> <p>Рік розробки: 2023</p> <p>Process returned 0 (0x0) execution time : 0.078 s</p> <p>Press any key to continue.</p>	passed
T_4	Вводим: ""	<p>Кількість ком: 0</p> <p>Кількість крапок: 0</p> <p>Дата та час дозапису: Thu May 04 04:18:38 2023</p> <p>""</p>	passed

		<p>Рік розробки: 2023</p> <p>Process returned 0 (0x0) execution time : 0.078 s</p> <p>Press any key to continue.</p>	
--	--	--	--

Висновок: У даній лабораторній роботі було розглянуто реалізацію програмних модулів оброблення даних складових типів з файловим введенням/виведенням у середовищі Code::Blocks IDE мовою C++. Було запропоновано завдання зчитування даних з файлу, визначення до якої мови належить текст, підрахунку кількості ком та крапок у тексті, виведення дати та часу дозапису, змінення порядку абзаців у тексті.

У процесі виконання лабораторної роботи я мав можливість набути практичних навичок у програмуванні на мові C++, роботі з файловим введенням/виведенням, використанні функцій стандартних бібліотек для оброблення символічної інформації.

Отже, виконання даної лабораторної роботи дозволить збагатити свої знання та навички у програмуванні на мові C++, що є важливим у подальшій роботі у галузі розробки програмного забезпечення.