Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

Кафедра кібербезпеки та програмного забезпечення Дисципліна:

БМТП

**Лабораторна робота №10 В-2**

**Тема:** **« Реалізація Програмних Модулів Оброблення даних складових типів з файловим введенням / виведенням »**

Виконав: ст. гр. КБ-24-1

Іванов Даніїл А.

Перевірив: викладач

Коваленко А.С.

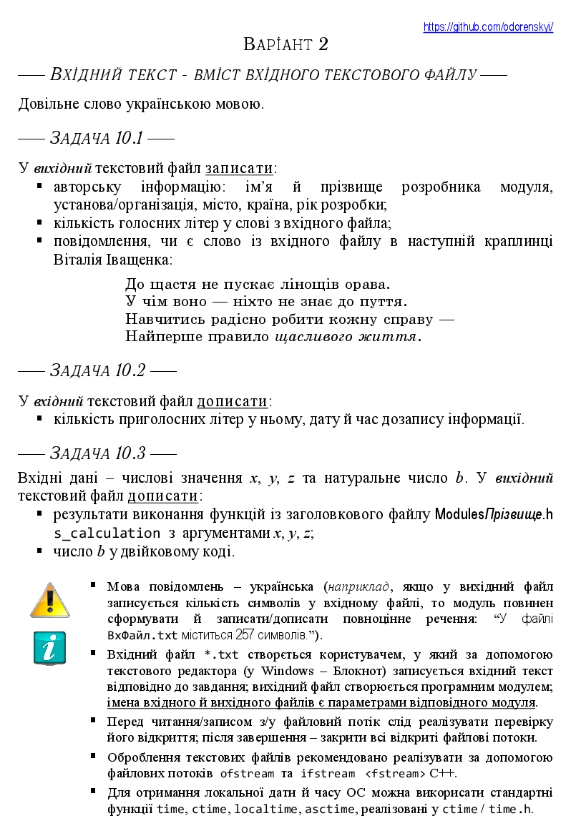
Кропивницький

2025

**ТЕМА: Реалізація Програмних Модулів Оброблення даних складових типів з файловим введенням / виведенням**

**МЕТА: полягає у набутті ґрунтовних вмінь і практичних навичок реалізації у Code::Blocks IDE мовою програмування С++ 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 символьної інформації.**

**ЗАВДАННЯ:**



Лістинг

#include

#include

#include

#include

#include

#include

#include <locale.h>

using namespace std;

// Функція підрахунку голосних літер int countVowels(const string& word) { setlocale(LC\_CTYPE,"ukr"); int count = 0; string vowels = "аеєиіїоуюяАЕЄИІЇОУЮЯ"; for (char ch : word) { if (vowels.find(ch) != string::npos) count++; } return count; }

// Функція підрахунку приголосних літер

int countConsonants(const string& word) { setlocale(LC\_CTYPE,"ukr");

int count = 0;

string vowels = "аеєиіїоуюяАЕЄИІЇОУЮЯ";

for (char ch : word) { if (isalpha(static\_cast(ch)) && vowels.find(ch) == string::npos) count++; }

return count; }

// Приклад функції обчислення

int s\_calculation(int x, int y, int z) {

return x \* y + z; // можна змінити на потрібну логіку }

int main() { setlocale(LC\_CTYPE,"ukr"); string word; ifstream fin("input.txt"); ofstream fout("output.txt");

if (!fin.is\_open() || !fout.is\_open()) {  
 cerr << "Не вдалося відкрити файли!" << endl;  
 return 1;  
}  
  
// Зчитування слова з вхідного файлу  
fin >> word;  
  
// Задача 10.1  
fout << "Автор: Іванов Даніїл Андрійович, ЦНТУ, Кропивницький, Україна, 2025\n";  
fout << "Кількість голосних літер: " << countVowels(word) << endl;  
  
// Перевірка наявності слова у вірші  
string poem[] = {  
 "До щастя не пускає лінощів орава.",  
 "У чім воно — ніхто не знає до пуття.",  
 "Навчитися радісно робити кожну справу —",  
 "Найперше правило щасливого життя."  
};  
  
bool found = false;  
for (string& line : poem) {  
 if (line.find(word) != string::npos) {  
 found = true;  
 break;  
 }  
}  
  
fout << "Слово \"" << word << "\" "  
 << (found ? "є" : "немає")  
 << " у вірші Віталія Іващенка.\n";  
  
// Задача 10.2  
int consonants = countConsonants(word);  
time\_t now = time(0);  
tm\* ltm = localtime(&now);  
fout << "Кількість приголосних літер: " << consonants << endl;  
fout << "Дата і час: " << 1900 + ltm->tm\_year << "-"  
 << 1 + ltm->tm\_mon << "-"  
 << ltm->tm\_mday << " "  
 << ltm->tm\_hour << ":" << ltm->tm\_min << ":" << ltm->tm\_sec << endl;  
  
// Задача 10.3  
int x = 3, y = 4, z = 5, b = 7;  
fout << "s\_calculation(" << x << "," << y << "," << z << ") = "  
 << s\_calculation(x, y, z) << endl;  
fout << "Число " << b << " у двійковому коді: " << bitset<8>(b) << endl;  
  
fin.close();  
fout.close();  
  
return 0;

}

Псевдокод

ПОЧАТОК

Встановити українську локаль

Оголосити функцію countVowels(слово):

Оголосити змінну лічильника голосних = 0

Оголосити рядок голосних літер

ДЛЯ кожного символу в слові:

Якщо символ є в голосних:

Збільшити лічильник

Повернути лічильник

Оголосити функцію countConsonants(слово):

Оголосити змінну лічильника приголосних = 0

Оголосити рядок голосних літер

ДЛЯ кожного символу в слові:

Якщо символ — літера і не входить до голосних:

Збільшити лічильник

Повернути лічильник

Оголосити функцію s\_calculation(x, y, z):

Повернути x \* y + z

Відкрити вхідний файл "input.txt"

Відкрити вихідний файл "output.txt"

Якщо файл не відкрився:

Вивести повідомлення про помилку

Завершити програму

Зчитати слово з файлу

ЗАПИСАТИ у файл інформацію про автора

// Задача 10.1

Обчислити кількість голосних у слові

Записати результат у файл

Оголосити масив з рядками вірша

Оголосити змінну "знайдено" = false

ДЛЯ кожного рядка у вірші:

Якщо слово знайдено у рядку:

встановити "знайдено" = true

вийти з циклу

Записати у файл, чи є слово у вірші

// Задача 10.2

Обчислити кількість приголосних

Отримати поточну дату і час

Записати кількість приголосних і час у файл

// Задача 10.3

Задати x = 3, y = 4, z = 5, b = 7

Обчислити s\_calculation(x, y, z)

Записати результат у файл

Записати b у двійковій формі у файл

Закрити вхідний і вихідний файли

КІНЕЦЬ

Аргументи

* Код структурований і поділений на функції.
* Є main() як точка входу, що відповідає стандарту C++.
* Логіка обробки розбита на задачі (10.1, 10.2, 10.3).
* Функції мають зрозумілі назви, що відповідають їхній меті.
* Використано функціональний підхід — повторне використання коду.
* Програма читає дані з вхідного файлу input.txt.
* Результати записуються у output.txt, що зручно для збереження виводу.
* Передбачено перевірку на успішне відкриття файлів.
* Використано ifstream/ofstream, що є стандартним для C++.
* Після завершення файли коректно закриваються.
* Програма працює з українською мовою.
* Враховано голосні та приголосні української абетки.
* Функції countVowels та countConsonants гнучкі і легко розширюються.
* Голосні літери перевіряються через find, що є ефективним способом.
* Приголосні визначаються як літери, які не є голосними
* Використано setlocale(LC\_CTYPE, "ukr") для підтримки українських літер.
* Локаль викликається у кожній функції для коректної обробки символів.
* Перевірка наявності слова у вірші виконується через find().
* Вірш збережено в масиві рядків, що дозволяє легко з ним працювати.
* Вивід результату у файлі зазначає, чи є слово у вірші.
* Програма виводить поточну дату та час.
* Використовується time() і localtime() для отримання системного часу.
* Вивід часу у форматі "YYYY-MM-DD HH:MM:SS", що зручно.
* Є функція s\_calculation, яка демонструє приклад обчислення.
* Можна змінити логіку s\_calculation для будь-яких формул.
* Аргументи передаються через змінні x, y, z.
* Значення b виводиться у двійковому форматі.
* Використано bitset<8>, що дозволяє зручно виводити число у бітовій формі.
* Бінарне представлення зручно для вивчення внутрішньої структури числа.
* Підходить для навчання роботі з файлами.
* Можна використовувати для лабораторних робіт у ЗВО.
* Вивчається робота з текстом і обробка символів.
* Знайомить із обробкою рядків та локалізацією.
* Код легко читається і зрозумілий для новачків.
* Можна легко модифікувати для інших мов.
* Функції універсальні — можна використовувати повторно.
* Код легко розширити новими функціями.
* Підтримує роботу з Unicode (частково, завдяки локалі).
* Є перевірка на відкриття файлів.
* Уникається робота з файлами, якщо вони не відкрилися.
* Немає ризиків переповнення буфера — використовується std::string.
* Можна винести вірш у зовнішній файл.
* Варто оптимізувати setlocale — викликати один раз.
* Варто використовувати std::wstring для повної підтримки Unicode.
* Можна додати обробку слів з кількох слів у рядку.
* Додати підрахунок слів, довжину, частоту літер.
* Зробити інтерфейс із параметрами командного рядка.
* Підключити логування дій програми.
* Застосувати регулярні вирази для точнішого пошуку у вірші.
* Створити тестовий набір для перевірки всіх функцій.