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”

РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ ОБРОБЛЕННЯ ДАНИХ СКЛАДОВИХ ТИПІВ З ФАЙЛОВИМ ВВЕДЕННЯМ/ВИВЕДЕННЯМ

ЗАВДАННЯ ВИДАВ

доцент кафедри кібербезпеки та програмного забезпечення

Доренський О. П.

[https://github.com/odorenskyi/](https://github.com/odorenskyi/Dmytro-Parkhomenko-KB18)

ВИКОНАВ

студент академічної групи КБ-24

Науменко О. В.

ПЕРЕВІРИВ

викладач кафедри кібербезпеки   
та програмного забезпечення

Коваленко А. С.

Кропивницький – 2025

Мета роботи 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:

1. Реалізувати програмні модулі розв’язування задач 10.1–10.3 як складові статичної бібліотеки libModulesПрізвище.а (проект ModulesПрізвище лабораторних робіт №8–9).
2. Реалізувати тестовий драйвер автоматизованої перевірки програмних модулів розв’язування задач 10.1–10.3.9.4 на основі функцій статичної бібліотеки libModulesПрізвище.а.

**Варіант №15:**

**ЗАДАЧА 10.1**

**Формулювання задачі:**

**У вихідний текстовий файл записати:**

-авторську інформацію: ім’я й прізвище розробника модуля, установа/організація, місто, країна, рік розробки

-кількість символів у вхідному файлі

-повідомлення, чи є у вхідному файлі слова “програма”, “модуль”, “студент”, “програміст” (у програмі слід реалізувати розрізнення слів “модуль” та “модульчик”, “студент” та “студентка”, “модуль” та “модульне” тощо)

**Вимоги до програмного забезпечення:**

-підтримка аналізу тексту українською мовою

-розпізнавання слів із коренями або частинами базових форм (стемінг або просте порівняння з використанням find/regex)

-перевірка наявності ключових слів

-підтримка формування текстового звіту з результатами аналізу

**Архітектура рішення:**

-input.txt — вхідний файл з довільним українським текстом

-output.txt — вихідний файл із результатами

-analyze\_text() — функція для обчислення кількості символів та пошуку ключових слів

-write\_output() — функція для створення текстового звіту

-contains\_word\_variants(word, text) — допоміжна функція, яка перевіряє наявність слова з можливими варіаціями

**Проєктування програмних модулів:**

mod\_text\_analysis.cpp

-int count\_characters(const std::string &text) - повертає кількість символів

-bool contains\_word(const std::string &text, const std::vector<std::string> &variants) - перевіряє наявність слова та його форм

-void write\_report(const std::string &filename, ...) - записує звіт

mod\_main.cpp

-зчитування вхідного тексту

-виклик модулів аналізу

-запис результату

**ЗАДАЧА 10.2**

**Формулювання задачі:**

**У вихідний текстовий файл дописати:**

-перелік Державних символів України згідно зі ст. 20 Конституції

України

-дату дописування інформації

**Вимоги до програмного забезпечення:**

-можливість додавання інформації у вже наявний файл

-автоматичне зчитування поточної дати

-можливість локалізованого виводу тексту

**Архітектура рішення:**

-output.txt — файл, який уже існує після Задачі 10.1

-append\_symbols() — функція для відкриття файлу в режимі

дописування та запису тексту

-get\_current\_date() — функція для отримання поточної дати в заданому

форматі

**Проєктування програмних модулів:**

mod\_constitution.cpp

-std::string get\_ukrainian\_symbols() - повертає перелік символів

-std::string get\_current\_date() - формує дату у форматі ДД.ММ.РРРР

-void append\_info(const std::string &filename) - виконує дописування у

файл

**ЗАДАЧА 10.3**

**Формулювання задачі:**

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

-результат виконання функції з заголовкового файлу

Modules/Прізвище.h — s\_calculation(x, y, z)

-число b у двійковому коді

**Вимоги до програмного забезпечення:**

-підключення зовнішнього заголовкового файлу з реалізацією функції

-підтримка математичних обчислень

-конвертація десяткового числа у двійкову систему

**Архітектура рішення:**

-Modules/Прізвище.h — файл із функцією s\_calculation(x, y, z)

-output.txt — файл, у який додається результат функції та двійкове

представлення b

-binary\_conversion() — функція для переведення числа у двійковий

вигляд

**Проєктування програмних модулів:**

mod\_math\_ops.cpp

-int s\_calculation(int x, int y, int z) - реалізація згідно з індивідуальним

варіантом

-std::string to\_binary(int number) - повертає рядок з двійковим

поданням

-void append\_math\_results(const std::string &filename, int x, int y, int z, int

b) - виконує обчислення та запис результату