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

Авраменко В. 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.

**Варіант №1**

**Завдання до лабораторної роботи**

1. Реалізувати програмні модулі розв’язування задач 10.1–10.3 як складові статичної бібліотеки libModulesAvramenko.а (проект ModulesAvramenko, створений під час виконання лабораторної роботи №8-9).
2. Реалізувати тестовий драйвер автоматизованої перевірки програмних модулів розв’язування задач 10.1–10.3.

# Задача 10.1

Мета задачі: створення вихідного текстового файлу, до якого потрібно записати:

* Авторську інформацію (ім’я, прізвище, установа, місто, країна, рік розробки).
* Повне речення, що містить кількість символів у вхідному файлі.
* Повідомлення про наявність ключових слів, зокрема – слів *"програма"*, *"модуль"*, *"студент"*, *"програміст"* із необхідним точним співпадінням (тобто виключення схожих, але не ідентичних слів типу "модульчик", "студентка", "модульне").

**Основні моменти аналізу:**

1. **Обробка вхідного тексту:**
   * Читання даних із вхідного текстового файлу.
   * Обхід усіх символів файлу, щоб обчислити загальну кількість символів (враховуючи пробіли, знаки пунктуації тощо).
2. **Пошук ключових слів:**
   * Необхідно реалізувати алгоритм пошуку, що враховує кордони слів. Наприклад, використання регулярних виразів або розбиття тексту на окремі слова допоможе уникнути плутанини між подібними варіантами (як-от "модуль" та "модульчик").
   * Перевірити наявність кожного з перелічених слів та сформувати відповідне повідомлення, яке повідомляє, чи міститься кожне слово у тексті.
3. **Формування вихідного повідомлення:**

* Створення повноцінних речень українською мовою для запису:
  + - Автоматично згенерованого повідомлення з авторською інформацією.
    - Речення про кількість символів (наприклад: "У файлі ВхФайл.txt міститься 257 символів.").
    - Повідомлення про наявність кожного ключового слова (якщо слово міститься – "слово X присутнє", інакше – "слово X відсутнє").

1. **Робота з файлами:**
   * Перед записом у вихідний файл перевірити, чи існує він. Якщо існує – знищити вміст, інакше – створити новий файл.
   * Забезпечити коректне відкриття файлів за допомогою об'єктів ifstream/ofstream, а після роботи – їх закрити.

# Задача 10.2

**Мета задачі:** дописування до існуючого вхідного текстового файлу:

* Кількості цифр, які містяться у файлі.
* Дати та часу виконання дописування.

**Основні моменти аналізу:**

1. **Читання файлу:**
   * Повторно прочитати вхідний файл для отримання актуального тексту.
   * Обхід символів для підрахунку цифр:

* Використати стандартну функцію (наприклад, isdigit()), щоб перевірити кожний символ.

1. **Отримання поточної дати та часу:**

* Використання стандартних функцій роботи з часом (наприклад, time(), localtime(), ctime()) для формування форматованої строки з поточною датою та часом.
* Переконатись, що отриманий формат повідомлення відповідає вимогам та записується українською мовою.

1. **Дописування до файлу:**
   * Відкриття вхідного файлу у режимі дозапису (append mode).
   * Запис нового тексту, який включає:
     + Речення, що вказує кількість цифр у файлі.
     + Речення з датою й часом дозапису.
   * Обов’язково перевірити відкриття потоку для дозапису перед його використанням.

# Задача 10.3

**Мета задачі:**

* Отримати числові значення x, y, z та натуральне число b.
* Викликати функцію s\_calculation (яка знаходиться в заголовковому файлі ModulesПрізвище.h) для обчислення з аргументами x, y, z.
* Дописати до вихідного текстового файлу результати роботи функції.
* Дописати до вихідного файлу число b у двійковому представленні.

**Основні моменти аналізу:**

1. **Обробка числових даних:**

* Забезпечити правильне введення або передавання значень x, y, z та b.
* Переконатися, що значення b є натуральним числом (щоб уникнути некоректних перетворень).

1. **Обчислювальний модуль:**

* Викликати функцію s\_calculation, яка розраховує певне значення на основі трьох вхідних чисел.
* Забезпечити отримання результату обчислення та його конвертацію у вигляд, придатний для запису у текстовий файл (наприклад, привести до рядкового формату).

1. **Перетворення числа у двійковий код:**

* Реалізувати функцію (наприклад, ConvertToBinary(int b)), яка за допомогою алгоритму послідовного ділення на 2 формує рядкове представлення числа b у двійковій системі числення.
* Перевірити коректність обчислень та правильний формат вихідного рядка.

1. **Запис результатів:**

* Результат виклику s\_calculation(x, y, z) та двійкове представлення числа b дописуються до вихідного файлу.
* Як і в попередніх задачах, переконатися, що файл відкритий для дозапису та після операції файл закритий.