Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 9

з навчальної дисципліни

“Базові методології та технології програмування”

РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУЖЕНИХ

ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

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 технології модульного програмування, застосування операторів С/С++ арифметичних, логічних, побітових операцій, умови, циклів та вибору під час розроблення статичних бібліотек, заголовкових файлів та програмних засобів у кросплатформовому середовищі Code::Blocks.

**Варіант №19**

**Завдання до лабораторної роботи**

1. Реалізувати функції розв’язування задач 9.1–9.3 як складових статичної бібліотеки libModulesAvramenko.а (проект ModulesAvramenko, створений під час виконання лабораторної роботи №8).
2. Реалізувати програмне забезпечення розв’язування задачі 9.4 на основі функцій статичної бібліотеки libModulesAvramenko.а

## **ЗАДАЧА 9.1**

### ****1. Аналіз і постановка задачі****

**Мета:** визначити ступінь хвилювання моря (у балах) за заданою висотою хвиль у метрах.

**Вхідні дані:** дійсне число – висота хвиль у метрах.

**Вихідні дані:** ціле число від 0 до 9 – бал хвилювання моря відповідно до таблиці.

### ****2. Аналіз вимог****

* Програма має обробляти значення висоти хвиль з плаваючою комою.
* Необхідна точна відповідність діапазонам згідно таблиці.
* Вивід має бути числом (0–9), опціонально з описом терміну (наприклад, “ROUGH”).

### ****3. Проєктування архітектури****

* Проста консольна програма.
* Один вхід – число з плаваючою крапкою.
* Умовна конструкція if-elif-else або switch.

### ****4. Детальне проектування модулів****

**Функція** int wave\_grade(float h):

* Приймає висоту хвилі.
* Повертає оцінку хвилювання моря (0–9).

## **ЗАДАЧА 9.2**

### ****1. Аналіз і постановка задачі****

**Мета:** перетворити температуру з Фаренгейта в Цельсій.

**Вхідні дані:** дійсне число – температура у градусах Фаренгейта.

**Вихідні дані:** дійсне число – температура у градусах Цельсія.

### ****2. Аналіз вимог****

* Просте обчислення за відомою формулою.
* Необхідна правильність розрахунку з урахуванням десяткової точності.

### ****3. Проєктування архітектури****

* Одна функція, що приймає та повертає .
* Консольна програма з виводом результату.

### ****4. Детальне проектування модулів****

**Функція** float fahrenheit\_to\_celsius(float t\_f):

* Реалізує формулу: .

## **ЗАДАЧА 9.3**

### ****1. Аналіз і постановка задачі****

**Мета:** залежно від значення біта D₁ (другий справа, тобто біт №1), або підрахувати кількість бінарних нулів, або суму одиниць у двійковому представленні.

**Вхідні дані:** ціле число N від 0 до 65535.

**Вихідні дані:**

* якщо біт D₁ == 1 → рахуємо **кількість нулів**;
* якщо D₁ == 0 → рахуємо **суму одиниць**.

### ****2. Аналіз вимог****

* Потрібна побітова операція для визначення значення біта.
* Потрібна бінарна репрезентація числа.
* Для підрахунку – використовуємо тернарний оператор.

### ****3. Проєктування архітектури****

* Отримання біта через побітову операцію: (N >> 1) & 1
* Далі залежно від значення – рахуємо або нулі, або одиниці в бінарному представленні (bin(N)).

### ****4. Детальне проектування модулів****

**Функція** int process\_binary(int N):

* bit = (N >> 1) & 1
* result = bin(N)[2:]
* Якщо bit == 1: рахуємо кількість '0' у result
* Інакше – суму одиниць: sum(int(b) for b in result)