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

Заріцкий В. А.

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 №9**

**Завдання до лабораторної роботи**

1. Реалізувати статичну бібліотеку модулів libModulesПрізвище C/C++, яка містить функцію розв’язування задачі 8.1.
2. Реалізувати програмне забезпечення розв’язування задачі 8.2 — консольний застосунок

# Аналіз і постановка задачі 9.1

Вхід: Кількість спожитої електроенергії (кВт\*год).

Вихід: Сума до сплати у гривнях за такими тарифами:

* До 150 кВтгод (включно): 130,843 коп./кВтгод.
* Від 150 до 800 кВтгод (включно): 241,945 коп./кВтгод.
* Понад 800 кВтгод: 534,047 коп./кВтгод.

План реалізації: написати функцію, яка приймає кількість кВт⋅год і за допомогою розгалужувальних операторів (if-else або тернарного оператора) визначає відповідний тариф та обчислює суму.

Лістинг задачі 9.1:

double calculateElectricityBill(double kWh) {

double rate;

if (kWh <= 150)

rate = 0.130843;

else if (kWh <= 800)

rate = 0.241945;

else

rate = 0.534047;

return kWh \* rate;

}

# Аналіз і постановка задачі 9.2

**Вхід:** Масив із 12 значень середньомісячної температури (°C).

**Вихід:** Середня температура за шкалою Цельсія та Фаренгейта.

План реалізації:

1. Обчислити середнє значення за Цельсієм.
2. Перевести його у Фаренгейти за формулою:

Лістинг задачі 9.2:

struct Temperature {

double celsius;

double fahrenheit;

};

Temperature calculateAverageTemperature(const double temps[12]) {

double sum = 0.0;

for (int i = 0; i < 12; i++) {

sum += temps[i];

}

double avgC = sum / 12.0;

Temperature result;

result.celsius = avgC;

result.fahrenheit = 32 + (9.0 / 5.0) \* avgC;

return result;

}

# Аналіз і постановка задачі 9.3

**Вхід:** Натуральне число N (від 0 до 5,740,500).

**Вихід:**

* Якщо 10-й біт () числа N рівний 0, – вивести кількість двійкових нулів.
* Інакше – кількість двійкових одиниць.

План реалізації:

1. За допомогою побітових операцій перевірте 10-й біт.
2. Для підрахунку нулів або одиниць використовуйте цикл та тернарний оператор «? : ».
3. Якщо N = 0, повертати 1.

Лістинг задачі 9.3:

int countBits(int N) {

if (N == 0) {

return 1;

}

bool bit10 = (N & (1 << 10)) != 0;

int count = 0;

while (N > 0) {

int bit = N & 1;

// Якщо bit10 == false – підраховуємо нулі, інакше одиниці.

count += (bit10 ? (bit == 1 ? 1 : 0) : (bit == 0 ? 1 : 0));

N >>= 1;

}

return count;

}

Тестові набори:

# ****Набір 1: «Нормальні» дані (Typical Cases)****

**Задача 9.1:**

**Вхід:** 100 кВт⋅год

**Очікуваний результат:**

* Тариф для ≤150 кВт⋅год: 130,843 коп./кВт⋅год
* Розрахунок: 100 ⋅ 0.130843 = **13.0843 грн** (якщо рахувати в гривнях, де 100 коп. = 1 грн)

**Задача 9.2:**

**Вхід:** Масив з 12 температур, наприклад:

double temps[12] = {0.0, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0, 11.0};

**Очікуваний результат:**

* Середня температура (°C): (0+1+2+…+11) / 12 = 5.5 °C
* Перетворення: 32 + (9.0/5) \* 5.5 ≈ **41.9 °F**

**Задача 9.3:**

**Вхід:** N = 100

* У двійковій формі: 100 → «1100100»
* Перевірка 10-го біта: для числа 100 10-й біт (індекс 10, якщо рахувати від 0) має значення 0.

**Очікуваний результат:**

* Оскільки 10-й біт = 0, потрібно підрахувати кількість нулів.
* «1100100» містить 4 нулі → **4**

# ****Набір 2: Граничні значення (Boundary Cases)****

**Задача 9.1:**

**Вхід:** 150 кВт⋅год

**Очікуваний результат:**

* 150 кВт⋅год потрапляє до першої категорії (до 150 включно)
* Розрахунок: 150 ⋅ 0.130843 = **19.62645 грн**

**Задача 9.2:**

**Вхід:** Масив з 12 температур з негативними та позитивними значеннями, наприклад:

double temps[12] = {-5.0, -3.0, 0.0, 5.0, 10.0, 15.0, 20.0, 18.0, 12.0, 7.0, 0.0, -2.0};

**Очікуваний результат:**

* Сума ≈ 77.0 → середнє ≈ 6.4167 °C
* Перетворення: 32 + (9.0/5) \* 6.4167 ≈ **43.55 °F**

**Задача 9.3:**

**Вхід:** N = 1024

* У двійковій формі: 1024 → «10000000000»
* 10-й біт (за індексацією від 0) встановлено (значення 1).

**Очікуваний результат:**

* Оскільки 10-й біт ≠ 0, підраховуємо кількість одиниць.
* У «10000000000» лише 1 одиниця → **1**

# ****Набір 3: Крайові/Екстремальні значення (Edge Cases)****

**Задача 9.1:**

**Вхід:** 1000 кВтгод

**Очікуваний результат:**

* Оскільки 1000 > 800, використовується тариф 0.534047
* Розрахунок: 1000 ⋅ 0.534047 = **534.047 грн**

**Задача 9.2:**

**Вхід:** Масив, де всі значення однакові, наприклад:

double temps[12] = {25.0, 25.0, 25.0, 25.0, 25.0, 25.0, 25.0, 25.0, 25.0, 25.0, 25.0, 25.0};

**Очікуваний результат:**

* Середня температура = 25 °C
* Перетворення: 32 + (9.0/5) \* 25 = 32 + 45 = **77 °F**

**Задача 9.3:**

**Вхід:** N = 0

* Особливий випадок: 0 у двійковій системі зазвичай представляється як "0".
* За умовою, якщо число 0, очевидно, що результат – кількість нулів;
* **Примітка:** у функцію була додана обробка N = 0 (повертається 1, оскільки 0 має один розряд, який дорівнює 0).

**Очікуваний результат:** **1**