

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 РОЗГАЛУДЖЕННЯ
ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ
ПРОЦЕСІВ

ЗАВДАННЯ ВИДАВ
доцент кафедри кібербезпеки
та програмного забезпечення
Доренський О. П.
<https://github.com/odorenskyi/>

ВИКОНАВ
студент академічної групи КН-24
Ковальова Єва

ПЕРЕВІРИВ
ст. викладач кафедри кібербезпеки
та програмного забезпечення
Коваленко Анастасія Сергіївна

ТЕМА: РЕАЛІЗАЦІЯ СТАТИСТИЧНИХ БІБЛОТЕК МОДУЛІВ ЛІНІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

МЕТА: полягає у набутті ґрунтовних вмінь і практичних навичок реалізації технології модульного програмування, застосування операторів C / C++ арифметичних, логічних, побітових операцій, умови, циклів та вибору під час розроблення статичних бібліотек, заголовкових файлів та програмних засобів у кросплатформовому середовищі Code::Blocks.

ВАРІАНТ 10

— ЗАДАЧА 9.1 —

Тариф на спожитий газ (за один місяць) є наступним: до 208 м^3 за місяць вартує 1,299 грн. за м^3 ; до 500 м^3 – 1,788 грн./ м^3 , понад 500 м^3 – 3,645 грн./ м^3 .

Вхід: об'єм використаного природного газу за місяць (м^3).

Вихід: сума до сплати у гривнях за спожитий газ.

— ЗАДАЧА 9.2 —

Вхід: розмір взуття у сантиметрах.

Вихід: відповідники розмірів взуття в системах України та Великобританії.



Таблиця відповідності розмірів взуття в системах США, Європи, України, та в сантиметрах

Велико-британія	США		ЄС	см	Україна
	чоловічі	жіночі			
1	1,5	2,5	33	20,5	–
1,5	2	3	33 2/3	21	–
2	2,5	3,5	34 1/3	21,5	–
2,5	3	4	35	22	–
3	3,5	4,5	36	22,5	–
4	4,5	5,5	36 2/3	23	35
4,5	5	6	37 1/3	23,5	36
5	5,5	6,5	38	24	36,5
5,5	6	7	39	24,5	37
6	6,5	7,5	39 2/3	25	38
6,5	7	8	40 1/3	25,5	38/39
7,5	8	9	41	26	40,5

Велико-британія	США		ЄС	см	Україна
	чоловічі	жіночі			
8	8,5	9,5	42	26,5	41
8,5	9	10	42 2/3	27	41,5
9	9,5	10,5	43 1/3	27,5	42
9,5	10	11	44	28	42/43
10	10,5	11,5	45	28,5	43
11	11,5	12,5	45 2/3	29	44
11,5	12	13	46 1/3	29,5	45
12	12,5	13,5	47	30	46
12,5	13	14	48	30,5	47
13	13,5	14,5	48 2/3	31	47,5
14	14,5	15,5	49 1/3	31,5	48
14,5	15	16	50	32	48,5

— ЗАДАЧА 9.3 —

Вхід: натуральне число N від 0 до 21359010.

Вихід: якщо біт D_{11} числа N рівний 1, кількість двійкових нулів у ньому, інакше — кількість двійкових одиниць*.

*під час підрахунку кількості бінарних 0 або 1 рекомендовано використати тернарний оператор « ? : ».

— ЗАДАЧА 9.4 —

За введеним користувачем символом “u” викликається `s_calculation()`, “t” – функція задачі 9.1, “r” – функція задачі 9.2, “e” – функція задачі 9.3; якщо користувач вводить інші символи, вони ігноруються, при чому видається звуковий сигнал про помилкове введення. Після цього, якщо користувач за запитом додатка вводить символ “w” або “i”, відбувається вихід з програми, інакше — виконання програми повторюється.



У випадку, якщо параметром і/або результатом функції є дані нестандартного типу (наприклад, складового), то такий **тип варто реалізувати у заголовковому файлі**.

Аналіз і постановка задач 9.1, 9.2 та 9.3

Завдання 9.1 - Калькулятор оплати за газ

Ця задача вимагає розрахунку оплати за газ на основі багаторівневих тарифів:

- До 208 м³: 1.299 грн за м³
- Від 208 до 500 м³: 1.788 грн за м³
- Понад 500 м³: 3.645 грн за м³

Завдання 9.2 - Конвертер розміру взуття

Це завдання вимагає перетворення вхідного розміру взуття у сантиметрах на еквівалентні розміри в українській та британській системах, використовуючи надану таблицю конвертації.

Завдання 9.3 - Аналіз двійкового числа

Це завдання просить написати програму, яка:

- Приймає натуральне число N (від 0 до 21359010)
- Якщо біт D11 числа N дорівнює 1, підрахувати кількість двійкових нулів
- В іншому випадку, підрахувати кількість двійкових одиниць

Аналіз вимог, проектування архітектури та програмних модулів для кожної задачі

Завдання 9.1 - Калькулятор оплати за газ

Аналіз вимог:

- Розрахувати суму до сплати за спожитий газ за місяць
- Використовувати трирівневу систему тарифікації
- Вхідними даними є об'єм використаного газу в м³
- Вихідними даними є сума до сплати в гривнях

Проектування модуля:

- Створення функції яка приймає один параметр - об'єм спожитого газу
- Реалізація логіки розрахунку з урахуванням усіх тарифних діапазонів
- Повернення обчисленої суми з точністю до копійок

Завдання 9.2 - Конвертер розміру взуття

Аналіз вимог:

- **Перетворити розмір взуття із сантиметрів у відповідники Українського та Британського стандартів**
- **Використовувати надану таблицю конвертації**
- **Вхідними даними є розмір взуття у сантиметрах**
- **Вихідними даними є відповідні розміри в системах України та Великобританії**

Проектування модуля:

- **Створення функції яка приймає один параметр - розмір у сантиметрах**
- **Реалізація механізму пошуку відповідників за таблицею конвертації**
- **Обробка неточних розмірів з округленням до найближчого стандартного значення**
- **Повернення знайдених відповідників у вигляді кортежу або структурованого об'єкту**

Завдання 9.3 - Аналіз двійкового числа

Аналіз вимог:

- **Прийняти натуральне число N в діапазоні від 0 до 21359010**
- **Перевірити значення біту D11 (11-й біт справа, починаючи з 0)**
- **Залежно від значення біту D11, підрахувати кількість нулів або одиниць**
- **Вхідними даними є ціле число N**
- **Вихідними даними є кількість нулів (якщо D11=1) або кількість одиниць (якщо D11=0)**

Проектування модуля:

- **Створення функції яка приймає один параметр - натуральне число**
- **Реалізація конвертації числа у двійковий формат**
- **Визначення значення біту D11**
- **Підрахунок кількості нулів чи одиниць залежно від умови**
- **Обробка крайових випадків (мале число з менш ніж 12 бітами)**
- **Повернення обчисленого результату**

Лістинг 9.1

```
double gas_payment(double volume) {
    if (volume < 0) {
        std::cout << "Такого значення бути не може" << std::endl;
        return 0.0;
    }
    double rate = 0.0;
    if (volume <= 208) {
        rate = 1.299;
    } else if (volume > 208 && volume <= 500) {
        rate = 1.788;
    } else {
        rate = 3.645;
    }
    return volume * rate;
}
```

Лістинг 9.2

```
void shoes(float sizes)
{
    if (sizes==20.5){
        cout << "Розмір в Україні - відсутній , в Великобританії - 1" << endl;
    }
    else if (sizes==21) {
        cout << "Розмір в Україні - відсутній , в Великобританії - 1,5" << endl;
    }
    else if (sizes==21.5) {
        cout << "Розмір в Україні - відсутній , в Великобританії - 2" << endl;
    }
    else if (sizes==22){
        cout << "Розмір в Україні - відсутній , в Великобританії - 2.5" << endl;
    }
    else if (sizes==22.5){
        cout << "Розмір в Україні - відсутній , в Великобританії - 3" << endl;
    }
    else if (sizes==23){
        cout << "Розмір в Україні - 35 , в Великобританії - 4" << endl;
    }
    else if (sizes==23.5){
        cout << "Розмір в Україні - 36 , в Великобританії - 4.5" << endl;
    }
    else if (sizes==24){
        cout << "Розмір в Україні - 36.5 , в Великобританії - 5" << endl;
    }
    else if (sizes==24.5){
```

```
        cout << "Розмір в Україні - 37 , в Великобританії - 5.5" << endl;
    }
    else if (sizes==25){
        cout << "Розмір в Україні - 38 , в Великобританії - 6" << endl;
    }
    else if (sizes==25.5) {
        cout << "Розмір в Україні - 38/39 , в Великобританії - 6.5" << endl;
    }
    else if (sizes==26){
        cout << "Розмір в Україні - 40.5 , в Великобританії - 7.5" << endl;
    }
    else if (sizes==26.5){
        cout << "Розмір в Україні - 41 , в Великобританії - 8" << endl;
    }
    else if (sizes==27){
        cout << "Розмір в Україні - 41.5 , в Великобританії - 8.5" << endl;
    }
    else if (sizes==27.5){
        cout << "Розмір в Україні - 42 , в Великобританії - 9" << endl;
    }
    else if (sizes==28){
        cout << "Розмір в Україні - 42/43, в Великобританії - 9.5" << endl;
    }
    else if (sizes==28.5){
        cout << "Розмір в Україні - 43 , в Великобританії - 10" << endl;
    }
    else if (sizes==29){
        cout << "Розмір в Україні - 44 , в Великобританії - 11" << endl;
    }
    else if (sizes==29.5){
        cout << "Розмір в Україні - 45 , в Великобританії - 11.5" << endl;
    }
    else if (sizes==30){
        cout << "Розмір в Україні - 46 , в Великобританії - 12" << endl;
    }
    else if (sizes==30.5){
        cout << "Розмір в Україні - 47 , в Великобританії - 12.5" << endl;
    }
    else if (sizes==31){
        cout << "Розмір в Україні - 47.5 , в Великобританії - 13" << endl;
    }
    else if (sizes==31.5){
        cout << "Розмір в Україні - 48 , в Великобританії - 14" << endl;
    }
    else if (sizes==32){
```

```

        cout << "Розмір в Україні - 48.5 , в Великобританії - 14.5" << endl;
    }
    else{
        cout << "Такого розміру немає" << endl;
    }
}

```

Лістинг 9.3

```

int binary_count(int N) {
    if (N < 0 || N > 21359010) {
        cout << "Число виходить за межі допустимого діапазону (0 - 21359010)" <<
endl;
        return 1;
    }

    string binary = "";
    int temp = N;
    while (temp > 0) {
        binary = (temp % 2 == 0 ? "0" : "1") + binary;
        temp /= 2;
    }

    while (binary.length() < 12) {
        binary = "0" + binary;
    }

    int ones = 0, zeros = 0;
    for (char bit : binary) {
        if (bit == '1') {
            ones++;
        } else {
            zeros++;
        }
    }

    if (binary.length() < 12) {
        cout << "Недостатньо бітів" << endl;
        return 1;
    }

    if (binary[binary.length() - 12] == '0') {
        cout << "Кількість двійкових нулів: " << zeros << endl;
    } else {
        cout << "Сума двійкових одиниць: " << ones << endl;
    }
}

```

```
    return 0;
}
```

Лістинг TestDriver

```
#include <iostream>
#include <windows.h>
#include "ModulesKovalova.h"

using namespace std;

void developer_information() {
    std::cout << "Ковальова Єва ©, Усі права захищені." << std::endl;
}

int main() {
    SetConsoleOutputCP(65001);
    SetConsoleCP(65001);

    developer_information();

    // Виклик gas_payment
    double volume;
    cout << "Введіть об'єм газу (м³): ";
    cin >> volume;
    cout << "Сума до сплати за газ (" << volume << " м³): "
         << gas_payment(volume) << " грн" << endl;

    // Виклик shoes
    float size;
    cout << "Введіть довжину стопи (см): ";
    cin >> size;
    cout << "Розмір взуття для довжини стопи " << size << " см:" << endl;
    shoes(size);

    // Виклик binary_count
    int n;

    cout << "\nВведіть число: ";
    cin >> n;

    int result = binary_count(n);
    if (result != 1) {
        cout << "Результат обробки числа: " << result << endl;
    }
}
```


Аналіз і постановка задачі 9.4

Задача 9.4 вимагає створення головного модуля програми `s_calculation()`, який інтегрує всі три попередні функції та забезпечує інтерфейс користувача з наступними характеристиками:

1. При введенні користувачем символу викликається функція `u s_calculation()`
2. При введенні користувачем символу викликається функція `f` (задача 9.1) для розрахунку оплати за газ.
3. При введенні символу викликається функція `г` (задача 9.2) для конвертації розміру взуття.
4. При введенні символу викликається функція `е` (задача 9.3) для аналізу двійкового числа.
5. Всі інші символи ігноруються з виведенням звукового сигналу про помилкове введення.
6. Після виконання будь-якої функції, якщо користувач вводить символ `"w"` або `"i"`, програма завершується.
7. В іншому випадку програма продовжує виконання (повертається до вибору функції).

Лістинг програми `Kovalova_task`

```
#include <iostream>
#include <cmath>
#include <windows.h>
#include "ModulesKovalova.h"

using namespace std;

void developer_information() {
    std::cout << "Ковальова Єва ©, Усі права захищені." << std::endl;
}

void task_selector(char option) {
    while (true) {
        if (option == 'u') {
            double x, y, z;
            std::cout << "Введіть x, y, z для s_calculation: ";
            std::cin >> x >> y >> z;
            double result = s_calculation(x, y, z);
```

```

        if (std::isnan(result)) {
            std::cout << "Результат: NAN (у не може дорівнювати 0)" << std::endl;
        } else {
            std::cout << "Результат: " << result << std::endl;
        }
    } else if (option == 't') {
        double volume;
        std::cout << "Введіть обсяг газу: ";
        std::cin >> volume;
        std::cout << "Ціна за газ: " << gas_payment(volume) << " грн" <<
std::endl;
    } else if (option == 'r') {
        float sizes;
        std::cout << "Введіть довжину стопи (см): ";
        std::cin >> sizes;
        shoes(sizes);
    } else if (option == 'e') {
        unsigned int N;
        std::cout << "Введіть число N: ";
        std::cin >> N;
        binary_count(N);
    } else {
        std::cout << '\a'; // Звуковий сигнал
        std::cout << "Невідома опція. Спробуйте ще раз." << std::endl;
    }
}

char exit_option;
std::cout << "Введіть 'w' для виходу або 'i' для продовження: ";
std::cin >> exit_option;
if (exit_option == 'w') {
    break;
} else if (exit_option == 'i') {
    std::cout << "Продовження роботи програми." << std::endl;
    std::cout << "Введіть опцію (u, t, r, e): ";
    std::cin >> option;
} else {
    std::cout << "Невідома опція. Спробуйте ще раз." << std::endl;
}
}

}

int main()
{
    SetConsoleOutputCP(65001);
    SetConsoleCP(65001);

```

```
developer_information();  
char option;  
std::cout << "Введіть опцію (u, t, r, e): ";  
std::cin >> option;  
task_selector(option);  
return 0;  
}
```

Висновок до лабораторної роботи

Виконання цієї лабораторної роботи дозволило поглибити знання та вдосконалити навички в реалізації програмних модулів з використанням статичних бібліотек, а також застосуванні операторів розгалуження та ітераційних процесів у мові програмування C++. Було розглянуто основні принципи модульного програмування, що дає змогу покращити структуру коду, спростити його підтримку та повторне використання в інших проєктах.

У процесі роботи була здійснена розробка та тестування функцій для розв'язання поставлених задач із використанням статичних бібліотек. Було створено відповідні заголовкові файли та файли реалізації, що містять необхідні функції. Важливою частиною виконання лабораторної роботи стало правильне підключення бібліотек та організація проєкту в середовищі Code::Blocks, що сприяє розумінню структури програм на основі модулів.

Окрему увагу було приділено операторам розгалуження, які дозволяють виконувати різні дії залежно від заданих умов. Це важливий аспект програмування, оскільки розгалуження широко застосовуються в розробці алгоритмів для обробки даних та прийняття рішень. Ітераційні процеси, які були використані при реалізації розв'язання задач, забезпечують ефективне повторення певних дій до виконання необхідної умови. Було розглянуто різні види циклів, їх особливості та оптимальні випадки використання.

Результатом виконаної роботи стало створення програмного забезпечення, яке коректно обробляє вхідні дані та видає правильні результати відповідно до поставлених умов. Код було протестовано на різних наборах даних, що дозволило переконатися у його коректності та стабільності.

Під час виконання лабораторної роботи виникали певні труднощі, зокрема, з правильним підключенням статичних бібліотек та налаштуванням проєкту в

середовищі розробки. Було необхідно уважно слідкувати за структурою каталогів, шляхами до файлів та правильним оголошенням функцій у заголовкових файлах. Також важливим аспектом стало тестування отриманих результатів та усунення помилок, що сприяло поглибленню навичок налагодження програм.

Ця лабораторна робота є важливим етапом у вивченні програмування, оскільки вона дає змогу зрозуміти основи організації коду в масштабних проєктах. Реалізація статичних бібліотек дозволяє створювати багаторазово використовувані компоненти, що спрощує розробку складних програмних систем. Робота також продемонструвала переваги використання модульного підходу, що сприяє зручному розподілу обов'язків у командній розробці.

Використання операторів розгалуження та ітераційних процесів є необхідним для створення ефективних алгоритмів, що виконують складні обчислення та приймають рішення на основі аналізу вхідних даних. У ході виконання завдань було закріплено теоретичні знання та отримано практичний досвід роботи з такими конструкціями.

Здобуті навички можна застосовувати не лише в рамках навчального процесу, а й у реальних проєктах, де необхідна ефективна організація коду та оптимізація процесів. Лабораторна робота дала змогу краще зрозуміти принципи структурного програмування та підготувала до використання більш складних підходів у майбутньому.