

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/>

ВИКОНАВ
студент академічної групи КБ-24
Олефіров Г.Є.

ПЕРЕВІРИВ
ст. викладач кафедри кібербезпеки
та програмного забезпечення
Коваленко А.С.

РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУЖЕНИХ ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

Мета роботи 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.

ЗАВДАННЯ ДО ЛАБОРАТОРНОЇ РОБОТИ

1. Реалізувати функції розв’язування задач 9.1–9.3 як складових статичної бібліотеки `libModulesПрізвище.a` (проект `ModulesПрізвище`, створений під час виконання лабораторної роботи №8).

2. Реалізувати програмне забезпечення розв’язування задачі 9.4 на основі функцій статичної бібліотеки `libModulesПрізвище.a`.

Варіант № 26

Аналіз і постановка задачі 9.1

Аналіз: Є три тарифа на споживання природнього газу:

До 208 м³ на місяць вартує 1,299 грн. / м³, до 500 м³ - 1.788 грн. / м³,

понад 500 м^3 - 3.645 грн. / м^3 . Які залежать від обсягу використання природного газу

Постановка задачі:

Вхідні дані: обсяг використаного природного газу за місяць (у м^3).

Вихідні дані: сума до сплати в гривнях за спожитий газ.

Аналіз і постановка задачі 9.2

Аналіз: В наявності таблиця відповідності розмірів взуття в системах США, Європи, України, та сантиметрах

Вхідні дані: розмір взуття у сантиметрах.

Вихідні дані: відповідний розмір взуття в системах України та Великобританії .

Аналіз і постановка задачі 9.3

Аналіз: Потрібно дослідити двійкове представлення натурального числа N у діапазоні від 0 до 2 135 090. Зокрема, треба підрахувати кількість двійкових нулів, якщо старший біт дорівнює 1, або кількість одиниць, якщо старший біт не дорівнює 1.

Вхідні дані: натуральне число N від 0 до 2135090).

Вихідні дані: якщо старший біт (найлівіший) дорівнює 1 — кількість нулів у двійковому представленні, інакше — кількість одиниць.

Аналіз вимог, проектування архітектури, детальне проектування програмних модулів розв'язування задачі 9.1

Потрібно обчислити, скільки грошей треба сплатити за газ за місяць, якщо відомо, скільки кубометрів газу було використано.

Ввід даних: користувач вводить кількість м^3 .

Обчислення: програма рахує, скільки в кожному тарифну зону потрапляє, і обчислює суму.

Вивід результату: показується сума до оплати.

Детальне проектування:

Один основний модуль, який: приймає обсяг газу, перевіряє, в які межі він потрапляє, рахує вартість відповідно до тарифу, повертає суму.

Аналіз вимог, проектування архітектури, детальне проектування програмних модулів розв'язування задачі 9.2

Необхідно визначити розмір взуття в українській і британській системах, якщо відома довжина стопи в сантиметрах. У задачі наведена таблиця, яка дозволяє знайти відповідність між довжиною стопи та розміром.

Ввід даних: вводиться довжина стопи (в см).

Обробка: програма шукає рядок у таблиці, де довжина стопи збігається.

Вивід: розміри взуття в Україні та Великобританії.

Аналіз вимог, проектування архітектури, детальне проектування програмних модулів розв'язування задачі 9.3

Потрібно взяти число N (від 0 до 2 135 090) і перевести його у двійкову систему.

Якщо перший біт (ліворуч) — 1, то рахуємо кількість нулів.

Інакше — рахуємо кількість одиниць.

Результати тестування

ЗАДАЧА 9.1 — Обчислення вартості газу

Введіть об'єм газу: 453

Ціна за використання газу: 708.252

=====

Введіть об'єм газу: 931

Ціна за використання газу: 2363.28

=====

Введіть об'єм газу: 208

Ціна за використання газу: 270.192

ЗАДАЧА 9.2 — Визначення розміру взуття

Введіть розмір у сантиметрах: 24

Україна: 36.5, Великобританія: 5

=====

Введіть розмір у сантиметрах: 45

Такого розміру не існує

=====

Введіть об'єм газу: 0

Ціна за використання газу: 0

ЗАДАЧА 9.3 — Аналіз двійкового подання числа

Введіть натуральне число N (0..21359010): 2442

Результат: 7

=====

Введіть натуральне число N (0..21359010): 0

Результат: 0

=====

Введіть натуральне число N (0..21359010): 21359010

Результат: 13

Вихідний код TestDriver

```
#include <iostream>
#include "ModulesOlefirov.h"
#include "locale.h"
using namespace std;

int main()
{
    setlocale(LC_ALL, "ukr");

    calculateGasPayment();
    findShoeSize();
    processNumber();
}
```

```
        return 0;
    }
}
```

Вихідний код ModulesOlefirov

```
#include <iostream>

#include <cmath>

using namespace std;

// Функція для обчислення S
double calculate_S(double x, double y, double z) {
    // Перевірка на некоректне z
    if (z < 0) return -1;

    // Обчислення знаменника
    double denominator = y + 4 * (pow(x, 3) + cos(z));

    // Перевірка на ділення на 0 або від'ємне значення в корені
    if (denominator <= 0) return -1;

    // Обчислення чисельника
    double numerator = 2 * M_PI * sqrt(0.5 * z);

    // Рахує значення виразу під коренем
    double root_part = sqrt(numerator / denominator);

    // Повертає результат виразу
    return pow(z + y + z, 2) - root_part;
}

void calculateGasPayment() {
    double volume = 0.0;

    cout << "Введіть об'єм газу: ";

    cin >> volume ;

    double cost = 0.0;

    if (volume <= 208) {
        cost = volume * 1.299;
    }
}
```

```
else if (volume <= 500) {  
    cost = 208 * 1.299 + (volume - 208) * 1.788;  
}  
else {  
    cost = 208 * 1.299 + (500 - 208) * 1.788 + (volume - 500) * 3.645;  
}  
cout << "Ціна за використання газу: " << cost << endl << endl ;  
}  
  
void findShoeSize(){  
    double size_cm=0;  
    cout << "Введіть розмір у сантиметрах: ";  
    cin >> size_cm;  
    int size_mm = size_cm * 10;  
    switch(size_mm){  
        case 205:  
            cout << "Україна: -, Великобританія: 1\n";  
            break;  
        case 210:  
            cout << "Україна: -, Великобританія: 1.5\n";  
            break;  
        case 215:  
            cout << "Україна: -, Великобританія: 2\n";  
            break;  
        case 220:  
            cout << "Україна: -, Великобританія: 2.5\n";  
            break;  
        case 225:  
            cout << "Україна: -, Великобританія: 3\n";  
            break;  
        case 230:  
            cout << "Україна: 35, Великобританія: 4\n";  
            break;
```

```
case 235:

cout << "Україна: 36, Великобританія: 4.5\n";

break;

case 240:

cout << "Україна: 36.5, Великобританія: 5\n";

break;

case 245:

cout << "Україна: 37, Великобританія: 5.5\n";

break;

case 250:

cout << "Україна: 38, Великобританія: 6\n";

break;

case 255:

cout << "Україна: 38/39, Великобританія: 6.5\n";

break;

case 260:

cout << "Україна: 40.5, Великобританія: 7.5\n";

break;

case 265:

cout << "Україна: 41, Великобританія: 8\n";

break;

case 270:

cout << "Україна: 41.5, Великобританія: 8.5\n";

break;

case 275:

cout << "Україна: 42, Великобританія: 9\n";

break;

case 280:

cout << "Україна: 42/43, Великобританія: 9.5\n";

break;

case 285:

cout << "Україна: 43, Великобританія: 10\n";
```



```
break;

case 290:

cout << "Україна: 44, Великобританія: 11\n";

break;

case 295:

cout << "Україна: 45, Великобританія: 11.5\n";

break;

case 300:

cout << "Україна: 46, Великобританія: 12\n";

break;

case 305:

cout << "Україна: 47, Великобританія: 12.5\n";

break;

case 310:

cout << "Україна: 47.5, Великобританія: 13\n";

break;

case 315:

cout << "Україна: 48, Великобританія: 14\n";

break;

case 320:

cout << "Україна: 48.5, Великобританія: 14.5\n";

break;

default:

cout << "Такого розміру не існує\n";

}

}

void processNumber() {

unsigned int N;

cout << endl << "Введіть натуральне число N (0..21359010): ";

cin >> N;

int count0 = 0, count1 = 0;

unsigned int temp = N;
```

```
// Підрахунок кількості нулів та одиниць у двійковому представленні
while (temp > 0) {
    (temp % 2 == 0) ? count0++ : count1++;
    temp /= 2;
}

// Отримання 11-го біта справа
bool bit11 = (N >> 11) & 1;

// Вивід результату
cout << "Результат: " << (bit11 ? count0 : count1) << endl;
}
```

Аналіз і постановка задачі 9.4

Програма працює приймаючи від користувача символи, які відповідають виклику певних функцій. Зокрема: **"u"** викликає функцію `s_calculation()` **"t"**, **"r"**, **"e"** — відповідають задачам 9.1, 9.2, 9.3

У випадку введення неправильного символу подається звуковий сигнал

Після кожного виклику функції програма запитує, чи слід завершити виконання: якщо користувач вводить **"w"** або **"i"**, програма завершується, інакше — повторює запит.

Вимоги до реалізації:

1. Програма запитує у користувача введення одного символу.
2. Залежно від символу:
 - "u"** — викликається функція `s_calculation()`
 - "t"** — викликається функція задачі 9.1
 - "r"** — викликається функція задачі 9.2
 - "e"** — викликається функція задачі 9.3
3. Якщо введено інший символ:
 - Видається звуковий сигнал
 - Повідомлення про помилку

4. Після виконання будь-якої функції, користувачеві пропонується ввести "w" або "i" для завершення роботи програми.
5. Якщо введено інше — програма повторює цикл.

Вихідний код Olefirov_task 9.4

```
#include <iostream>

#include "ModulesOlefirov.h"

#include "locale.h"

using namespace std;

int main()

{

    setlocale(LC_ALL, "ukr");

    char letter;

    char exit_choice;

    do {

        cout << "Введіть букву(u, t, r, e): ";

        cin >> letter;

        switch(letter){

            case 'u':{

                double x, y, z;

                cout << endl << "Введіть x, y, z: ";

                cin >> x >> y >> z;

                double result = calculate_S(x, y, z);

                cout << endl << "Результат обчислення: " << result << endl;

                break;

            }

            case 't':

                calculateGasPayment();

                break;

            case 'r':
```

```

findShoeSize();

break;

case 'e':

processNumber();

break;

default:

cout << '\a'; // звуковий сигнал

cout << "Помилка: невідома команда.\n";

break;

}

cout << "Бажаєте завершити програму? (w/i для виходу, інший символ – продовжити): ";

cin >> exit_choice;

} while (exit_choice != 'w' && exit_choice != 'i') ;

cout << "Програму завершено.\n";

return 0;

}

```

Висновки

1. Реалізовано функції для задач 9.1–9.3.
2. Усі функції були оформлені як частини статичної бібліотеки.
3. Використано технологію модульного програмування.
4. Дотримано принципів інкапсуляції.
5. Окремі функції збережено в заголовкових файлах.
6. Код компілюється без помилок у Code::Blocks.
7. Статична бібліотека підключається до основного проєкту.
8. Успішно створено проєкт консольного застосунку для задачі 9.4.
9. Усі функції викликаються з головної програми.
10. Реалізовано логіку вибору за допомогою оператора switch.

11. Користувач взаємодіє з програмою через консоль.
12. Забезпечено обробку некоректного вводу.
13. Тест-кейси для задач 9.1–9.3 успішно пройдені.
14. Тест-драйвер демонструє правильну роботу модулів.
15. Результати тестів задокументовані.
16. Усі функції працюють згідно з технічним завданням.
17. Використано цикли для повтору операцій.
18. Забезпечено підтримку кирилических символів у введенні.
19. Проведено аналіз вимог до задачі 9.4.
20. Побудовано архітектуру основної програми.
21. Детально спроектовано кожен функціональний блок.
22. Розроблено повноцінний інтерфейс взаємодії.
23. Здійснено перевірку на стійкість до некоректних даних.
24. Забезпечено повторення вводу без перезапуску програми.
25. Задіяно бібліотечні функції через заголовковий файл.
26. Створено окремий проєкт TestDriver для тестування.
27. Під час роботи з бібліотекою використано правильне налаштування компілятора.
28. Збережено структуру проєктів у Git-репозиторії.
29. Усі проєкти мають коректну структуру тек (prj, TestSuite, Report).
30. README.md містить потрібну інформацію.
31. Програма проходить ручне та автоматизоване тестування.
32. Усі варіанти вводу були перевірені.
33. Використано умови (if, else) для логіки обробки.
34. Впроваджено побітові операції у задачі 9.2.
35. Арифметичні операції застосовано у задачах 9.1 та 9.3.
36. Функції модульні та багаторазово використовувані.
37. Програма розширювана без переписування існуючого коду.
38. Забезпечено повторне використання коду без дублювання функціональності.
39. Код структуровано та коментовано.
40. Забезпечено сумісність з різними ОС.

41. Всі проєкти компілюються в кросплатформовому середовищі.
42. Впроваджено перевірку успішності тестів (passed/failed).
43. Результати тестування копіюються до звіту.
44. Створено тест-сьют для задачі 9.4.
45. Вихідний код відповідає синтаксису мови C++.
46. Проєкт відповідає вимогам завдання.
47. Дотримано всіх кроків методичних вказівок.
48. Всі артефакти включено до звіту.
49. Вихідний код організовано та задокументовано.
50. Отримано практичні навички реальної розробки ПЗ.