

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 РОЗГАЛУДЖЕНИХ ТА  
ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

ЗАВДАННЯ ВИДАВ

доцент кафедри кібербезпеки  
та програмного забезпечення

Доренський О. П.

<https://github.com/odorenskyi/>

ВИКОНАВ

студент академічної групи  
КІ-23

Чепіль В.О.

ПЕРЕВІРИВ

ст. викладач  
кафедри кібербезпеки  
та програмного забезпечення  
Дреєва Г. М.

Кропивницький – 2024

## **ТЕМА: РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУДЖЕНИХ ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ**

**МЕТА:** Набути 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:**

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 №18**

## ВАРІАНТ 18

### — ЗАДАЧА 9.1 —

За кожну покупку на суму до 1000 грн. нараховується 1% знижки на всю суму покупки; від 1000 до 2000 грн. – 5%; від 2000 до 3000 грн. – 7%; від 3000 до 5000 грн. – 10%, від 5000 до 10000 грн. – 15%, понад 10000 грн. сплачується 80% вартості.

Вхід: сума покупки.

Вихід: сума до сплати (із врахуванням знижки).

### — ЗАДАЧА 9.2 —

Вхід: розмір чоловічої спідні за українською системою.

Вихід: відповідники розмірів чоловічої білизни у системах Франції, Великобританії та міжнародній системі.



Таблиця відповідності розмірів чоловічої білизни

Міжнародний	Чехія, Словаччина	Україна	Велико- британія	Німеччина	Франція
S	6	44	34	–	2
M	7	46	36	4	3
L	8	48	38	5	4
XL	9	50	40	7	5
XXL	10	52	42	7	6
XXXL	–	54	44	8	–

### — ЗАДАЧА 9.3 —

Вхід: натуральне число  $N$  від 0 до 80000.

Вихід: якщо біт  $D_5$  числа  $N$  рівний 0, кількість двійкових нулів у ньому, інакше — суму двійкових одиниць\*.

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

### — ЗАДАЧА 9.4 —

За введеним користувачем символом “s” викликається `s_calculation()`, “f” – функція задачі 9.1, “g” – функція задачі 9.2, “h” – 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 “J” або “j”, відбувається вихід з програми, інакше — виконання програми повторюється.



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

### **Аналіз і постановка задачі 9.1:**

Вх: score – типу float(для точності обчислення)

Вих: expense – типу float(для точності обчислення)

Особливості:

score > 0 (Так як сума покупки не може бути від'ємна або = 0)

Діапазон:

до 1000 грн. – знижка становить 1%;

від 1000 до 2000 грн. – знижка становить 5%;

від 2000 до 3000 грн. – знижка становить 7%;

від 3000 до 5000 грн. – знижка становить 10%;

від 5000 до 10000 грн. – знижка становить 15%;

понад 10000 грн. – покупець сплачує 80% від вартості, тобто отримує 20% знижку.

Архітектура:

Є функцією модуля (discount\_calculation), що приймає аргумент типу float, обчислює змінну score та на базі даних змінної повертає вивід у потоці cout

### **Алгоритм до задачі 9.1:**

Початок

1) Введення суми покупки(змінна score)

- 2) Перевірка введених даних через умову (if-else)
- 3) Фільтрування вибору знижки через умови(if)
- 4) Обрахунок знижки від суми покупки(змінна expense)
- 5) Обчислення суми до сплати(score -= expense)
- 6) Виведення суми до сплати

Кінець

### **Аналіз і постановка задачі 9.2:**

Вх: underwear\_size - типу short int

Вих: underwear\_size буде виводити відповідні розміри

Особливості:

Коли вводимуться непарні числа та числа  $> 54$  і числа  $< 44$ , буде виведено «Таких розмірів немає!».

Архітектура:

Є функцією модуля (matching\_sizes), що приймає аргумент типу char, та повертає розмір через порівняння в діапазоні значень за заданою таблицею.

### **Алгоритм до задачі 9.2:**

Початок

- 1) Введення розміру білизни(змінна underwear\_size)

2) Перевірка введених даних через умову (if-else)

3) Вибір розміру за допомогою умов (if)

4) Виведення розміру білизни

Кінець

### **Аналіз і постановка задачі 9.3:**

Вх: N - змінна типу int

Вих: "Кількість двійкових нулів: <binary\_zero>" чи "Кількість двійкових одиниць: <binary\_one>"

Особливості:

N має бути в діапазоні від 0 до 80000

Вивід того чи іншого рядка залежить від біту числа, а саме D5, якщо він 0, то вивід: "Кількість двійкових нулів: <binary\_zero>", в іншому випадку: "Суму двійкових одиниць: <sum\_one>"

Архітектура:

Є функцією модуля (bit\_range), що приймає аргумент типу int, обчислює змінну binary та на базі даних змінної повертає вивід у потоці cout

### **Алгоритм до задачі 9.3:**

Початок

1) Встановити параметр функції зі значенням N.

2) У випадку введення невірних даних, вивести відповідне повідомлення через cout та завершити виконання програми.

- 3) Конвертувати вказане число в бінарний формат і зберегти його як рядкову змінну.
- 4) Визначити та занотувати кількість нулів і одиниць у бінарному рядку, використовуючи цілочисельні змінні.
- 5) Переконатися, що в числі присутні рівно 5 бітів; якщо це не так, вивести відповідне повідомлення через cout та припинити роботу програми.
- 6) За допомогою вибору користувача визначити, які дані (0 або 1) потрібно вивести, і показати результат через потік cout.

Кінець

#### **Аналіз і постановка задачі 9.4:**

Вх: symbol - змінна типу char

Вих: Виконання однієї з функцій на вибір.

Особливості:

Користувач може вводити символи 's', 'f', 'g', 'h', 'J', 'j', які представлені типом char, для вибору відповідної функції.

У разі введення символу, що не відповідає жодному з дозволених запитів, програма повинна повторно запросити ввід та відтворити звуковий сигнал, сигналізуючи про помилкове введення.

Архітектура:

Проект передбачає розробку консольного додатку, який включатиме раніше розроблений модуль. Основний зміст проекту полягає у забезпеченні механізму виклику функцій згідно з технічним завданням, виходячи з символу, введенного користувачем.



## Алгоритм до задачі 9.4:

Початок

- 1) Вивести запрошення до введення команди через потік cout.
- 2) Запропонувати користувачу ввести символ для визначення команди.
- 3) Використати вибір з декількох опцій для ідентифікації команди.
- 4) Вивести повідомлення, яке вимагає введення додаткових даних відповідно до вибраної команди.
- 5) У разі введення даних, що не відповідають вимогам, активувати звуковий сигнал та розпочати виконання програми спочатку; якщо все вірно – обробити запит.

Кінець

## Лістинг ModulesChepil.h

```
#include <iostream>

#include <cmath>

#include <iomanip>

#include <string>

#include <bits/stdc++.h>

using namespace std;

void s_calculation(int x, int y, int z)

{

    system("chcp 65001 > nul");
```

```

const double PI = 3.141592653589793238;

float S;

S = pow(2 * z + 1, x) - sqrt(abs(y - (1 / 2) * z)) + z + PI;

cout << "S: " << setprecision(1) << fixed << S << endl;

}

void discount_calculation(float score)

{

    system("chcp 65001 > nul");

    float expense;

    if(score > 0){

        if(score < 1000){

            expense = score * 0.01;

        }

        if(score >= 1000 && score < 2000){

            expense = score * 0.05;

        }

        if(score >= 2000 && score < 3000){

            expense = score * 0.07;

        }

        if(score >= 3000 && score < 5000){

            expense = score * 0.1;

        }

        if(score >= 5000 && score < 10000){

            expense = score * 0.15;

        }

    }

}

```

```

if(score >= 10000){

    expense = score * 0.2;

}

score -= expense;

cout << "Сума до сплати: " << setprecision(2) << fixed << score << endl;

}

else{

    cout << "Введені не відповідні дані" << endl;

}

}

void matching_sizes(short int underwear_size)

{

    system("chcp 65001 > nul");

    if (underwear_size%2 == 0 && (underwear_size <= 54 && underwear_size >= 44)){

        if (underwear_size == 44) {

            cout << "Міжнародний розмір: S " << endl;

            cout << "Чехія, Словаччина розмір: 6 " << endl;

            cout << "Великобританія розмір: 34 " << endl;

            cout << "Німеччина розмір: немає " << endl;

            cout << "Франція розмір: 2 " << endl;

        }

        if (underwear_size == 46) {

            cout << "Міжнародний розмір: M " << endl;

            cout << "Чехія, Словаччина розмір: 7" << endl;

            cout << "Великобританія розмір: 36 " << endl;

        }

    }

}

```

```
cout << "Німечина розмір: 4 " << endl;

cout << "Франція розмір: 3 " << endl;

}

if (underwear_size == 48) {

    cout << "Міжнародний розмір: L " << endl;

    cout << "Чехія, Словаччина розмір: 8 " << endl;

    cout << "Великобританія розмір: 38 " << endl;

    cout << "Німечина розмір: 5 " << endl;

    cout << "Франція розмір: 4 " << endl;

}

if (underwear_size == 50) {

    cout << "Міжнародний розмір: XL " << endl;

    cout << "Чехія, Словаччина розмір: 9 " << endl;

    cout << "Великобританія розмір: 40 " << endl;

    cout << "Німечина розмір: 7 " << endl;

    cout << "Франція розмір: 5 " << endl;

}

if (underwear_size == 52) {

    cout << "Міжнародний розмір: XXL " << endl;

    cout << "Чехія, Словаччина розмір: 10" << endl;

    cout << "Великобританія розмір: 42 " << endl;

    cout << "Німечина розмір: 7 " << endl;

    cout << "Франція розмір: 6 " << endl;

}

if (underwear_size == 54) {
```

```

        cout << "Міжнародний розмір: XXXL " << endl;

        cout << "Чехія, Словаччина розмір: немає " << endl;

        cout << "Великобританія розмір: 44 " << endl;

        cout << "Німеччина розмір: 8 " << endl;

        cout << "Франція розмір: немає " << endl;

    }

}

else {

    printf("Таких розмірів немає\n");

}

}

int bit_range(int N)

{

    system("chcp 65001 > nul");

    if (N < 0 || N > 80000) {

        cout << "Число не входить в діапазон <0 - 80000>" << endl;

        return 1;

    }

    int cou = 0, k = 1;

    while (N)

    {

        cou += (N%2) * k;

        k *= 10;

        N/=2;

```

```

    }

    string str = to_string(cou);

    int sum_one = count(str.begin(), str.end(), '1');

    int binary_zero = count(str.begin(), str.end(), '0');

    reverse(str.begin(), str.end());

    if (str.length() < 6) {

        cout << "Недостатньо бітів" << endl;

        return 1;

    }

    switch (str[5]) {

        case '0':

            cout << "Кількість двійкових нулів: " << binary_zero << endl;

            break;

        case '1':

            cout << "Сума двійкових одиниць: " << sum_one << endl;

            break;

    }

    return 0;

```

## Лістинг TestDriver

```

#include <iostream>

#include <ModulesChepil.h>

using namespace std;

```

```
int main()

{

    system("chcp 65001 & cls");

    cout << "Задача 9.1\nTC-1 ";

    discount_calculation(-100);

    cout << "TC-2 ";

    discount_calculation(0);

    cout << "TC-3 ";

    discount_calculation(500);

    cout << "TC-4 ";

    discount_calculation(1500);

    cout << "Tc-5 ";

    discount_calculation(2200);

    cout << "TC-6 ";

    discount_calculation(4300);

    cout << "TC-7 ";

    discount_calculation(8000);

    cout << "TC-8 ";

    discount_calculation(15000);

    cout << "TC-9 ";

    discount_calculation(125.50);

    cout << "TC-10 ";

    discount_calculation(735.00);


    cout << "\nЗадача 9.2\nTC-1\n";
```

```
matching_sizes(44);

cout << "\nTC-2\n";

matching_sizes(46);

cout << "\nTC-3\n";

matching_sizes(48);

cout << "\nTC-4\n";

matching_sizes(50);

cout << "\nTC-5\n";

matching_sizes(52);

cout << "\nTC-6\n";

matching_sizes(54);

cout << "\nTC-7\n";

matching_sizes(43);

cout << "\nTC-8\n";

matching_sizes(55);

cout << "\nTC-9\n";

matching_sizes(-10);

cout << "\nTC-10\n";

matching_sizes(0);

cout << "\n";


cout << "Задача 9.3\nTC-1 ";

bit_range(20);

cout << "TC-2 ";

bit_range(25);
```



```
    cout << "TC-3 ";  
  
    bit_range(30);  
  
    cout << "TC-4 ";  
  
    bit_range(35);  
  
    cout << "TC-5 ";  
  
    bit_range(70);  
  
    cout << "TC-6 ";  
  
    bit_range(200);  
  
    cout << "TC-7 ";  
  
    bit_range(1000);  
  
    cout << "TC-8 ";  
  
    bit_range(777);  
  
    cout << "TC-9 ";  
  
    bit_range(96);  
  
    cout << "TC-10 ";  
  
    bit_range(158);  
  
}
```

### **Лістинг Chepil\_task**

```
#include <iostream>  
  
#include <ModulesChepil.h>  
  
  
using namespace std;  
  
  
int main()
```

```
{  
  
    system("chcp 65001 > nul");  
  
    char symbol;  
  
    cout << "Введіть символ виклику функції: " << endl;  
  
    cin >> symbol;  
  
    switch (symbol) {  
  
        case 's':  
  
            int x, y, z;  
  
            cout << "Введіть x:" << endl;  
  
            cin >> x;  
  
            cout << "Введіть y:" << endl;  
  
            cin >> y;  
  
            cout << "Введіть z:" << endl;  
  
            cin >> z;  
  
            s_calculation(x, y, z);  
  
            break;  
  
        case 'f':  
  
            int score;  
  
            cout << "Введіть суму покупки:" << endl;  
  
            cin >> score;  
  
            discount_calculation(score);  
  
            break;  
  
        case 'g':  
  
            short int underwear_size;  
  
            cout << "Введіть розмір білизни:" << endl;
```

```

        cin >> underwear_size;

        matching_sizes(underwear_size);

        break;

case 'h':

    int N;

    cout << "Введіть N:" << endl;

    cin >> N;

    bit_range(N);

    break;

case 'J':

    exit(1);

    break;

case 'j':

    exit(1);

    break;

default:

    cout << "\a\a" << endl;

}

main();

}

```

**Висновок:** під час виконання лабораторної роботи було виконано:

В рамках завдання 9.1, я провів аналіз вимог програмного забезпечення, розробив функцію для розрахунку знижок (discount\_calculation) і включив її до статичної бібліотеки libModulesCheril.a, де вже була функція для обчислення s (s\_calculation).

Також оновив заголовочний файл ModulesChepil.h, додавши прототип функції s\_calculation.

У виконанні завдання 9.2 я аналізував вимоги програмного забезпечення, створив функцію для визначення відповідності розмірів (matching\_sizes) і додав її до libModulesChepil.a, де вже присутні функції s\_calculation та boff. Після цього оновив заголовочний файл ModulesChepil.h, включивши прототипи функцій s\_calculation та discount\_calculation.

При виконанні завдання 9.3, було проведено аналіз вимог програмного забезпечення, розроблена функція для визначення діапазону бітів (bit\_range) і додана в статичну бібліотеку libModulesChepil.a, що тепер включає функції s\_calculation, discount\_calculation, matching\_sizes. Заголовочний файл ModulesChepil.h був оновлений з прототипами цих функцій.

У рамках завдання 9.4 я проаналізував вимоги до програмного забезпечення, створив консольний додаток Chepil\_task, до якого імпортував статичну бібліотеку libModulesChepil.a, що містить функції s\_calculation, discount\_calculation, matching\_sizes, bit\_range. Оновлений заголовочний файл ModulesChepil.h із прототипами всіх цих функцій також був включений.

Було створено 3 модульних та 1 системний TestSuites з 10 тест-кейсами в кожному.

### **Відповіді на контрольні запитання:**

1. Яким оператором C/C++ можливо повноцінно замінити тернарний оператор? Відповідь обґрунтуйте й доведіть експериментально.

Тернарний оператор можна повноцінно замінити оператором if-else. Тернарний оператор ?: дозволяє вибирати між двома значеннями в залежності від логічного виразу, подібно до того, як оператор if-else дозволяє виконувати різні блоки коду.

Наприклад, `int x = (умова) ? a : b;` може бути переписано як `int x; if (умова) x = a; else x = b;`.

2. Що в програмуванні розуміють під пріоритетом виконання операцій та асоціативністю?

Пріоритет виконання операторів визначає порядок, у якому виконуються операції в складних виразах, не враховуючи дужки. Асоціативність визначає порядок, у якому оператори того самого пріоритету виконуються – зліва направо чи справа наліво.

3. Яку область видимості мають змінні, оголошені в тілі циклу або умови (вибору)? Відповідь обґрунтуйте та доведіть експериментально.

Змінні, оголошені в тілі циклу або умови, мають блочну область видимості. Це означає, що вони існують і доступні лише всередині блоку, де були оголошені. Наприклад, якщо змінну оголосити всередині циклу `for`, вона не буде доступна поза цим циклом.

4. Якою є асоціативність операцій арифметичних, логічних, логічних порозрядних, інкрементна, декрементна, тернарної та порівняння в мові програмування C/C++?

Асоціативність більшості арифметичних та логічних операцій в C/C++ - зліва направо. Винятком є оператори присвоєння та унарні оператори (включаючи інкремент (++), декремент (--)), асоціативність яких справа наліво. Тернарний оператор має праву асоціативність.

5. Перелічіть випадки, за яких доцільно використовувати тернарний оператор C/C++, й наведіть приклад його запису.

Тернарний оператор доцільно використовувати для простих умовних присвоєнь або коли потрібно зробити код коротшим і читабельнішим. Наприклад: `int result = (a > b) ? a : b;` - це короткий спосіб присвоїти `result` значення `a`, якщо `a > b`, інакше `b`.

6. Яке значення міститиме змінна `cnt` після виконання наступної інструкції: `cnt--;` ?

Змінна `cnt` зменшиться на 1 після виконання інструкції `cnt--;`. Якщо перед цим `cnt` було, наприклад, 5, то після буде 4.

7. Чим константна змінна, оголошена за допомогою кваліфікатора типів `const`, відрізняється від змінної? Сформулюйте правило, коли змінну варто оголошувати саме константною.

Константна змінна не може бути змінена після її ініціалізації. Змінну варто оголошувати константною, коли відомо, що її значення не повинно змінюватися протягом часу виконання програми, щоб запобігти помилкам і зробити код безпечнішим.

8. Яких типів можуть бути операнди логічних операторів C/C++?

Операндами логічних операторів у C/C++ можуть бути будь-які вирази, що можуть бути оцінені як булеві значення (`true` або `false`), включно з числовими типами даних, де 0 оцінюється як `false`, а будь-яке інше значення - як `true`.

9. Яке значення міститиме змінна `cnt` при: `bool cnt = !!0;` ?

Змінна `cnt` буде мати значення `false`, оскільки `!!0` еквівалентно `false`. Оператор `!` інвертує значення, тому подвійне використання повертає вихідне значення, але як булевий тип.

10. Сформулюйте правило запису виразу ініціалізації у циклах з параметром (`for`) C++.

Правило запису ініціалізації у циклах `for` в C++ виглядає так: `for (ініціалізація; умова; інкремент/декремент) { тіло циклу }`. Ініціалізація зазвичай використовується для задання початкового значення лічильнику. Наприклад, `for (int i = 0; i < 10; i++) { /* тіло циклу */ }`.