

Міністерство освіти і науки України  
Центральноукраїнський національний технічний університет  
Механіко-технологічний факультет  
Кафедра кібербезпеки та програмного забезпечення  
Дисципліна: Базові методології та технології програмування

**Лабораторна робота №9**  
**Тема: «РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУЖЕНИХ  
ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ»**

Виконав: ст. гр. КН-24  
Куріщенко П. В.  
Перевірів: викладач  
Коваленко А.С.

Кропивницький 2025

## Варіант - 4

*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`.

### ПОРЯДОК ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ

#### Завдання 1.

(9.1)

##### Аналіз вимог:

Вхідні дані: Сума покупки(дійсне число).

Вихідні дані: Розмір знижки у відсотках та гривнях(ціле та дійсне число), сума до сплати(дійсне число).

**ОДЗ:** `purc_sum > 0`

##### Результати тестування `ffun_cal_discount`:

Enter the purchase amount: -10

Error: entered value less than one

-----

```
Enter the purchase amount: 200
Your discount in percent is 7%
Your discount in hryvnia is 14 uah
Your purchase amount with discount is 186
```

```
-----
Enter the purchase amount: 10000
Your discount in percent is 25%
Your discount in hryvnia is 2500 uah
Your purchase amount with discount is 7500
```

(9.2)

### **Аналіз вимог:**

Вхідні дані: Розмір білизни в українській системі(ціле число).

Вихідні дані: Розмір білизни в системах Франції, Великобританії та Міжнародній системі.

**ОДЗ:** uw\_size = {44, 46, 48, 50, 52, 54}

### **Результати тестування sfun\_male\_sizes:**

```
Enter men's underwear size (44 - 54) in Ukrainian system: 32
Error: entered a non-existing men's underwear size (44 - 54) in the
Ukrainian system
```

```
-----
Enter men's underwear size (44 - 54) in Ukrainian system: 48
Your size in International system: L
Your size in UK system: 38
Your size in French system: 4
```

```
-----
Enter men's underwear size (44 - 54) in Ukrainian system: 54
Your size in International system: XXXL
Your size in UK system: 44
Your size in French system: -
```

(9.3)

### **Аналіз вимог:**

Вхідні дані: Натуральне число від 0 до 190550.

Вихідні дані: якщо третій біт числа `number == 1`, кількість двійкових нулів у ньому, інакше — кількість двійкових одиниць.

**ОДЗ:** `0 >= number <= 190550`

### Результати тестування `tfun_bin_count`:

```
Enter a natural number from 0 to 190550: -23
```

```
Error: value entered is out of range
```

```
-----
```

```
Enter a natural number from 0 to 190550: 11022
```

```
The third bit is one, then the number of zeros is 11
```

```
-----
```

```
Enter a natural number from 0 to 190550: 190550
```

```
The third bit is zero, then the number of ones is 9
```

### Вихідний код (TestDriver):

```
#include <iostream>
```

```
#include "sample_module.h"
```

```
using namespace std;
```

```
void ffun_cal_discount(float purc_sum);
```

```
void sfun_male_sizes(int uw_size);
```

```
void tfun_bin_count(int number);
```

```
int main()
```

```
{
```

```
    float purc_sum = 0;
```

```
    cout << "Enter the purchase amount: ";
```

```
    cin >> purc_sum;
```

```
    ffun_cal_discount(purc_sum);
```

```
    int uw_size = 0;
```

```
    cout << "Enter men's underwear size (44 - 54) in Ukrainian  
system: ";
```

```

cin >> uw_size;

sfun_male_sizes(uw_size);

int number = 0;
cout << "Enter a natural number from 0 to 190550: ";
cin >> number;

tfun_bin_count(number);

return 0;
}

```

### Вихідний код (ModulesKurishchenko):

```

#include "sample_module.h"

float s_calculator(float x, float y, float z)
{
    float S;

    if (x <= y) {
        cout << "Error: log(x - y) is undefined (x must be greater
than y).\n";
    }
    else if (x + (z / 2) * pow(y, 2) == 0) {
        cout << "Error: Division by zero.\n";
    }
    else {
        S = log(x - y) + sqrt((M_PI * pow(x, 2)) / (x + (z / 2 *
pow(y, 2))));
        cout << "Your result is " << S;
    }
}

void ffun_cal_discount(float purc_sum) {

```

```

int discount = 0;
if (purc_sum <= 0){
    cout << "Error: entered value less than one\n";
    return;
}
if (purc_sum <= 100) discount = 5;
else if (purc_sum <= 200 && purc_sum > 100) discount = 7;
else if (purc_sum <= 300 && purc_sum > 200) discount = 9;
else if (purc_sum <= 400 && purc_sum > 300) discount = 12;
else if (purc_sum <= 500 && purc_sum > 400) discount = 15;
else if (purc_sum <= 1000 && purc_sum > 500) discount = 20;
else discount = 25;
cout << "Your discount in percent is " << discount << "%\n";
cout << "Your discount in hryvnia is " << discount * purc_sum /
100 << " uah\n";
cout << "Your purchase amount with discount is " << purc_sum -
(discount * purc_sum / 100) << "\n";
}

```

```

void sfun_male_sizes(int uw_size){
    switch (uw_size){
        case 44:
            cout << "Your size in International system: S\n";
            cout << "Your size in UK system: 34\n";
            cout << "Your size in French system: 2\n";
            break;
        case 46:
            cout << "Your size in International system: M\n";
            cout << "Your size in UK system: 36\n";
            cout << "Your size in French system: 3\n";
            break;
        case 48:
            cout << "Your size in International system: L\n";
            cout << "Your size in UK system: 38\n";
            cout << "Your size in French system: 4\n";
            break;
        case 50:

```

```

        cout << "Your size in International system: XL\n";
        cout << "Your size in UK system: 40\n";
        cout << "Your size in French system: 5\n";
        break;
    case 52:
        cout << "Your size in International system: XXL\n";
        cout << "Your size in UK system: 42\n";
        cout << "Your size in French system: 6\n";
        break;
    case 54:
        cout << "Your size in International system: XXXL\n";
        cout << "Your size in UK system: 44\n";
        cout << "Your size in French system: -\n";
        break;
    default:
        cout << "Error: entered a non-existing men's underwear
size (44 - 54) in the Ukrainian system\n";
        return;
    }
}

```

```

void tfun_bin_count(int number){
    if (number < 0 || number > 190550){
        cout << "Error: value entered is out of range\n";
        return;
    }
    bitset<18> binary(number);
    bool D3 = binary[3];
    int bin_amount = D3 ? (18 - binary.count()) : binary.count();
    cout << "The third bit is " << (D3 ? "one" : "zero")
        << ", then the number of " << (D3 ? "zeros" : "ones")
        << " is " << bin_amount << "\n";
}

```

```

void Dev_info()
{
    cout << "+-----+\n";
}

```

```
cout << "| Pavlo Kurishchenko(c) |\n";  
cout << "| github.com/movavok |\n";  
cout << "+-----+\n";  
}
```

(9.4)

#### **Аналіз вимог:**

Вхідні дані: Символи : s, g, h, f - для виклику функцій та J, j - для завершення програми.

#### **Висновок:**

Під час виконання лабораторної роботи №9 я закріпив свої навички програмування, аналізу алгоритмів та тестування. Зокрема, я навчився:

1. Аналізувати умову задачі та визначати ключові параметри для розв'язання.
2. Використовувати умовні оператори для реалізації різних сценаріїв роботи програми.
3. Працювати з таблицями відповідності для перетворення даних між різними системами.
4. Застосовувати тернарний оператор для скорочення коду та підвищення його читабельності.
5. Реалізовувати алгоритми обчислення знижок залежно від діапазону значень.
6. Конвертувати розміри чоловічої білизни між різними міжнародними стандартами.
7. Виконувати бітові операції для аналізу окремих розрядів числа.
8. Підраховувати кількість двійкових нулів та одиниць у числі.
9. Реалізовувати структури керування для виклику функцій за введеним символом.
10. Створювати механізми обробки помилкових введень користувача.



11. Використовувати звуковий сигнал для оповіщення про неправильне введення.

12. Впроваджувати механізм виходу з програми за бажанням користувача.

13. Організовувати структуру коду для зручного розширення та модифікації.

14. Використовувати функціональний підхід до написання програм.

15. Виконувати декомпозицію задачі на підзадачі та реалізовувати їх окремими функціями.

16. Впроваджувати перевірку коректності введених даних.

17. Працювати з різними типами даних та їх перетворенням.

18. Використовувати цикли для обробки даних.

19. Розуміти важливість коментарів у коді та писати їх правильно.

20. Використовувати модульний підхід до програмування.

21. Реалізовувати трасувальну таблицю для перевірки роботи алгоритму.

22. Створювати тестові кейси для перевірки коректності реалізації задачі.

23. Визначати критерії оцінки правильності розв'язку.

24. Аналізувати логічну правильність виконаних операцій.

25. Реалізовувати вхідні перевірки для запобігання помилкам у роботі програми.

26. Використовувати таблиці відповідності для конвертації значень.

27. Оптимізувати код для зменшення його розміру та підвищення ефективності.

28. Виконувати обробку виняткових ситуацій.

29. Формувати коректний та структурований вивід даних.

30. Використовувати змінні для збереження та обробки даних.

31. Розуміти принципи побудови діалогових інтерфейсів у консольних програмах.

32. Аналізувати та використовувати довідкові матеріали для розв'язання задач.

33. Створювати програми, що реагують на дії користувача.
34. Впроваджувати системи перевірки та корекції введених даних.
35. Розрізняти та застосовувати логічні оператори для контролю виконання програми.
36. Формулювати зрозумілі повідомлення про помилки.
37. Використовувати базові алгоритми для роботи з числами та рядками.
38. Реалізовувати та тестувати різні варіанти розрахунку знижок.
39. Вивчати та застосовувати альтернативні методи розв'язку задачі.
40. Оптимізувати програму, зменшуючи кількість виконуваних операцій.
41. Використовувати відладку для пошуку та виправлення помилок у коді.
42. Виконувати тестування програми на граничних значеннях.
43. Використовувати комбінації умов для перевірки складних випадків.
44. Визначати найефективніші способи представлення даних у програмі.
45. Аналізувати часову складність алгоритмів.
46. Вивчати та застосовувати концепції структурного програмування.
47. Реалізовувати механізм багаторазового виконання програми.
48. Використовувати стандартні функції для роботи з рядками та числами.
49. Розрізняти глобальні та локальні змінні та їхній вплив на виконання програми.
50. Розуміти принципи побудови ефективних алгоритмів.
51. Виконувати перевірку правильності вхідних та вихідних даних.
52. Використовувати оператори умовного вибору для реалізації логіки програми.
53. Працювати з різними типами циклів для виконання повторюваних завдань.
54. Використовувати функціональний підхід до розробки програмного забезпечення.

Отже, виконання цієї лабораторної роботи допомогло мені значно розширити знання та навички у сфері програмування, алгоритмізації та роботи з даними.