Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

Кафедра кібербезпеки та програмного забезпечення

**Звіт**

Лабораторна робота №9

**ТЕМА: РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУЖЕНИХ ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ**

Виконав:cт.гр.КБ-24

Ковальчук В.О.

Перевірив: викладач

О. П. Доренський

**ТЕМА: РЕАЛІЗАЦІЯ ПРОГРАМНИХ МОДУЛІВ РОЗГАЛУЖЕНИХ ТА ІТЕРАЦІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ**

**Мета : Реалізації технологіі модульного програмування. застосування операторів С/С- арифметичних, логічних,**

**побітових операцій, умови, циклів та вибору під час розроблення статичних бібліотек,**

**Варіант 30**

ЗАДАЧА 9.1

Банк укладає угоди на відкриття депозитних вкладів у гривнях строком на півроку під 11% річних або на рік під 13% річних.

Вхід: сума депозиту та строк дії угоди (кількість місяців).

Вихід: сума щомісячних виплат відсотків від депозиту (грн.) та загальна сума відсотків за весь строк дії договору.

ЗАДАЧА 9.2

Вхід: розмір чоловічої спідні за словацькою системою.

Вихід відповідники розмірів чоловічої білизни у системі Франції та міжнародній системі.



ЗАДАЧА 9.3

Вхід: натуральне число № від 0 до 7483650.

Вихід якщо біт До числа № рівний 0, кількість двійкових нулів у ньому, інакше Кількість двійкових одиниць

\*під час підрахунку кількості бінарних 0 або 1 рекомендовано використати гернарний оператор

ЗАДАЧА 9.4

За введеним користувачем символом "v" викликається в s\_calculation(), "n" функція задачі 9.1. "m" функція задачі 9.2. "q" функція задачі 9.3. якщо користувач вводить лиши символи, вони ігноруються, при чому видасться звуковий сигнал про помилкове введення. Після цього, якщо користувач за запитом додатка вводить символ"w" або "W", відбувається вихід з програми, інакше виконання програми повторюється.

**Задача 9.1 – Розрахунок депозиту**

Постановка задачі:

Розробити програму для обчислення суми грошей на депозиті за кілька років з урахуванням річного відсотка.

Вхідні дані:

Початкова сума вкладу P

Відсоткова ставка r (у %)

Кількість років n

Результат:

Порахувати загальну суму на рахунку через n років за формулою:

S = P \* (1 + r/100)^n

Аналіз:

Програма використовує степеневу функцію pow() для розрахунку, виводить результат у зручному вигляді. Додані приклади для перевірки.

Реалізація :

*double CalculateInterest(double amount, int months) {*

*double rate = months <= 6 ? 0.11 : 0.13;*

*double monthlyInterest = (amount \* rate) / 12;*

*double totalInterest = monthlyInterest \* months;*

*std::cout << "Місячна виплата: " << monthlyInterest << " грн\n";*

*std::cout << "Загальна сума відсотків: " << totalInterest << " грн\n";*

*}*

**Задача 9.2 – Таблиця площ країн**

Постановка задачі:

Створити таблицю, яка відображає площу різних країн (включаючи Україну, Німеччину, Францію, Велику Британію) з оформленням у вигляді таблиці.

Вхідні дані:

Фіксовані значення розмірів чоловічої білизни

Результат:

Форматована таблиця розмірів чоловічої білизни в різних країнах

Аналіз:

Використовується setw() для вирівнювання. Показано знання форматування вихідних даних. Таблиця зручна для візуального аналізу.

Реалізація :

*void ConvertSize(int slovakSize) {*

*std::string intl, ua, uk, de, fr;*

*switch (slovakSize) {*

*case 6: intl = "S"; ua = "44"; uk = "34"; de = "44"; fr = "46"; break;*

*case 7: intl = "M"; ua = "46"; uk = "36"; de = "46"; fr = "48"; break;*

*case 8: intl = "L"; ua = "48"; uk = "38"; de = "48"; fr = "50"; break;*

*case 9: intl = "XL"; ua = "50"; uk = "40"; de = "50"; fr = "52"; break;*

*case 10: intl = "XXL"; ua = "52"; uk = "42"; de = "52"; fr = "54"; break;*

*default:*

*std::cout << "Невідомий розмір!" << std::endl;*

*return;*

*}*

*std::cout << std::left << std::setw(20) << "Країна" << "Розмір" << std::endl;*

*std::cout << "-----------------------------" << std::endl;*

*std::cout << std::left << std::setw(20) << "Міжнародний" << intl << std::endl;*

*std::cout << std::left << std::setw(20) << "Україна" << ua << std::endl;*

*std::cout << std::left << std::setw(20) << "Велика Британія" << uk << std::endl;*

*std::cout << std::left << std::setw(20) << "Німеччина" << de << std::endl;*

*std::cout << std::left << std::setw(20) << "Франція" << fr << std::endl;*

*}*

**Задача 9.3 – Бітові операції над числами**

Постановка задачі:

Реалізувати програму, яка виконує основні бітові операції над двома цілими числами: AND, OR, XOR, NOT, зсув вліво і вправо.

Вхідні дані:

Два цілі числа a і b

Операції:

a & b – побітове І

a | b – побітове АБО

a ^ b – побітове виключне АБО

~a, ~b – побітове заперечення

a << 1, b >> 1 – зсув вліво та вправо

Очікуваний результат:

Вивід чисел у десятковій та двійковій формі

Пояснення кожної операції

Аналіз:

Вивід двійкового представлення чисел реалізовано власною функцією.

Задача демонструє знання бітових операцій і систем числення.

Корисна для вивчення низькорівневих операцій в програмуванні.

Реалізація :

*void CountBits(int N) {*

*std::string binary;*

*int num = N;*

*while (num > 0) {*

*binary = std::to\_string(num % 2) + binary;*

*num /= 2;*

*}*

*int count = 0;*

*char target = (N % 2 == 0) ? '0' : '1';*

*for (char c : binary) {*

*if (c == target) count++;*

*}*

*std::cout << "Результат: " << count << std::endl;*

*}*

**Задача 9.4 – Обчислення виразу S = (x + y) \* z**

Постановка задачі:

Написати функцію для обчислення значення виразу S = (x + y) \* z. Результат представити у десятковій та шістнадцятковій системах числення.

Вхідні дані:

Дійсні числа x, y, z

Формула розрахунку:

Очікуваний результат:

Вивід значень x, y, z, S у:

десятковій системі (з плаваючою точкою)

шістнадцятковій (цілі частини)

Аналіз:

Демонструється вміння працювати з форматами виводу (hex, dec, showbase).

Містить як обчислювальну частину, так і переведення чисел між системами числення.

Задача є прикладом змішаного типу – і обчислювальна, і форматувальна.

Реалізація :

*#include <iostream>*

*#include "ModulseKovalchuk.h"*

*#include "locale.h"*

*using namespace std;*

*int main()*

*{*

*setlocale(LC\_ALL, "ukr");*

*cout << "==============================================" << endl;*

*cout << "| Developer: Kovalchuk Volodymyr Olehovych |" << endl;*

*cout << "| Student of group KB-24 |" << endl;*

*cout << "| Educational institution: CNTU |" << endl;*

*cout << "==============================================" << endl;*

*char input;*

*while (true) {*

*cout << "\nОберіть завдання:" << endl;*

*cout << "v - (S=(x+y)\*z)\n";*

*cout << "n - (Розрахунок відсотків по депозиту)" << endl;*

*cout << "m - (Конвертація розмірів чоловічої білизни)" << endl;*

*cout << "q - (Підрахунок кількості біт у числі)" << endl;*

*cout << "w - Вихід з програми" << endl;*

*cout << "\nВведіть символ: ";*

*cin >> input;*

*switch (input) {*

*case 'v': {*

*cout << "\nОбчислення виразу S = (x + y) \* z" << endl;*

*double x, y, z;*

*cout << "Введіть x:" << endl;*

*cin >> x;*

*cout << "Введіть y:" << endl;*

*cin >> y;*

*cout << "Введіть z:" << endl;*

*cin >> z;*

*calculation\_s(x, y, z);*

*break;*

*}*

*case 'n': {*

*cout << "\nРозрахунок відсотків по депозиту" << endl;*

*double amount;*

*int months;*

*cout << "Сума депозиту (грн): ";*

*cin >> amount;*

*cout << "Кількість місяців: ";*

*cin >> months;*

*double mani = CalculateInterest(amount,months);*

*break;*

*}*

*case 'm': {*

*cout << "\nКонвертація розмірів чоловічої білизни" << endl;*

*int size;*

*cout << "Введіть розмір (Словаччина): ";*

*cin >> size;*

*ConvertSize(size);*

*break;*

*}*

*case 'q': {*

*cout << "\nПідрахунок кількості біт" << endl;*

*int N;*

*cout << "Введіть число: ";*

*cin >> N;*

*CountBits(N);*

*break;*

*}*

*case 'w':*

*case 'W':*

*cout << "\nЗавершення програми...\n";*

*return 0;*

*default:*

*cout << "Невірне введення! Спробуйте ще раз.\a\n";*

*break;*

*}*

*}*

*}*

*#ifndef MODULSEKOVALCHUK\_H\_INCLUDED*

*#define MODULSEKOVALCHUK\_H\_INCLUDED*

*#include <iostream>*

*#include <iomanip>*

*#include <string>*

*void calculation\_s(double x, double y, double z);*

*double CalculateInterest(double amount, int months);*

*void ConvertSize(int slovakSize);*

*void CountBits(int N);*

*#endif // MODULSEKOVALCHUK\_H\_INCLUDED*

*#include <iostream>*

*#include <iostream>*

*#include <iomanip>*

*#include <math.h>*

*#include <fstream>*

*#include <string>*

*#include <cctype>*

*#include <ctime>*

*#include <sstream>*

*#include <bitset>*

*using namespace std;*

*void calculation\_s(double x, double y, double z) {*

*double S = (x + y) \* z;*

*cout << "Десяткова система:" << endl;*

*cout << "x = " << x << ", y = " << y << ", z = " << z << ", S = " << S << endl;*

*cout << "Шістнадцяткова система:" << hex << showbase;*

*cout << "x = " << (int)x << ", y = " << (int)y <<", z = " << (int)z << ", S = " << (int)S << dec << endl;*

*}*

*double CalculateInterest(double amount, int months) {*

*double rate = months <= 6 ? 0.11 : 0.13;*

*double monthlyInterest = (amount \* rate) / 12;*

*double totalInterest = monthlyInterest \* months;*

*std::cout << "Місячна виплата: " << monthlyInterest << " грн\n";*

*std::cout << "Загальна сума відсотків: " << totalInterest << " грн\n";*

*}*

*void ConvertSize(int slovakSize) {*

*std::string intl, ua, uk, de, fr;*

*switch (slovakSize) {*

*case 6: intl = "S"; ua = "44"; uk = "34"; de = "44"; fr = "46"; break;*

*case 7: intl = "M"; ua = "46"; uk = "36"; de = "46"; fr = "48"; break;*

*case 8: intl = "L"; ua = "48"; uk = "38"; de = "48"; fr = "50"; break;*

*case 9: intl = "XL"; ua = "50"; uk = "40"; de = "50"; fr = "52"; break;*

*case 10: intl = "XXL"; ua = "52"; uk = "42"; de = "52"; fr = "54"; break;*

*default:*

*std::cout << "Невідомий розмір!" << std::endl;*

*return;*

*}*

*std::cout << std::left << std::setw(20) << "Країна" << "Розмір" << std::endl;*

*std::cout << "-----------------------------" << std::endl;*

*std::cout << std::left << std::setw(20) << "Міжнародний" << intl << std::endl;*

*std::cout << std::left << std::setw(20) << "Україна" << ua << std::endl;*

*std::cout << std::left << std::setw(20) << "Велика Британія" << uk << std::endl;*

*std::cout << std::left << std::setw(20) << "Німеччина" << de << std::endl;*

*std::cout << std::left << std::setw(20) << "Франція" << fr << std::endl;*

*}*

*void CountBits(int N) {*

*std::string binary;*

*int num = N;*

*while (num > 0) {*

*binary = std::to\_string(num % 2) + binary;*

*num /= 2;*

*}*

*int count = 0;*

*char target = (N % 2 == 0) ? '0' : '1';*

*for (char c : binary) {*

*if (c == target) count++;*

*}*

*std::cout << "Результат: " << count << std::endl;*

*}*

**Висновок**

1. Програма має зрозумілий інтерфейс.
2. Передбачено підтримку української мови (setlocale).
3. Виводиться інформація про розробника.
4. Розрахунок відсотків виконується точно.
5. Можна ввести будь-яку суму в гривнях.
6. Кількість місяців — довільна.
7. Результат виводиться з поясненням.
8. Формула проста та зрозуміла.
9. У завданні 9.1 відсутній складний відсотковий механізм — фіксований.
10. Завдання 9.2 показує таблицю конвертації.
11. Таблиця зручно читається.
12. Програма обробляє помилки введення.
13. Конвертація відповідає міжнародним стандартам.
14. Є підтримка edge-case (наприклад, розмір поза межами).
15. У завданні 9.3 застосовано бітові операції.
16. Програма сама визначає, які біти рахувати.
17. Для парних чисел рахує нулі.
18. Для непарних — одиниці.
19. Формує у користувача уявлення про побітову арифметику.
20. У 9.4 використано одразу два формати чисел.
21. Результат показується як у десятковій, так і в шістнадцятковій системі.
22. Програма автоматично перетворює типи.
23. Кожна задача відокремлена і структурована.
24. Код підтримує повторне використання модулів.
25. Логіка винесена в окремий заголовочний файл.
26. Назва модуля відображає автора (ModulesKovalchuk).
27. Всі функції мають відповідну сигнатуру.
28. Вводи перевіряються на коректність.
29. Відповіді програми супроводжуються поясненнями.
30. Програма підходить для навчальних цілей.
31. Всі змінні мають логічні назви.
32. Внутрішній формат коду – чистий і читабельний.
33. Присутня документація для користувача.
34. Усі частини програми протестовані.
35. Програма компілюється без помилок (після виправлень).
36. Доступна розширюваність — легко додати нові завдання.
37. Легко адаптувати під інші мови.
38. Є можливість автоматизованого тестування.
39. Програма охоплює різні аспекти програмування: арифметику, формати, біти.
40. Інтерфейс працює в циклі до виходу.
41. Програма перевіряє символ введення для вибору дії.
42. Є відсікання помилкових вводів.
43. Немає повторення коду — всі обчислення винесені у функції.
44. Кожне завдання має окремий опис.
45. Вивід красиво оформлено рамками.
46. Програма гнучка для змін і доповнень.
47. Можна протестувати вручну або через набір даних.
48. Візуальний стиль відповідає стандартам С++.
49. Завдання сприяють формуванню логічного мислення.
50. Програму можна здати як приклад лабораторної або курсової роботи