МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

ЗВІТ

ПРО ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ № 9

з навчальної дисципліни

“Реалізація програмних модулів розгалужених та ітераційних обчислювальних процесів”

На тему:

«Програмна Реалізація статичних бібліотек модулів лінійних обчислювальних процесів»

ВИКОНАВ

студент академічної КБ-23

Ліннікова Софія

ПЕРЕВІРИВ

викладач кафедри кібербезпеки

та програмного забезпечення

Дрєєва Г.М

Кропивницький – 2024

**Тема:** Реалізація програмних модулів розгалужених та ітераційних обчислювальних процесів

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

**Варіант № 15**

**Завдання 9.1**

Згідно з трудовим договором працівникові нараховується надбавка до заробітної плати за стаж роботи на підприємстві наступним чином: від 3 до 5 років стажу 10% від посадового окладу, від 5 до 7 - 15%; від 7 до 15 - 20%, понад 15 - 25%. Вхід: розмір посадового окладу (грн.), розмір заробітної плати працівника (грн.), стаж роботи на підприємстві (років). Вихід: сума до виплати (грн., включно з надбавкою).

**Алгоритм**

1. Ввести розмір посадового окладу, розмір заробітної плати працівника та його стаж роботи на підприємстві.
2. Визначити, в який діапазон стажу входить працівник.
3. Розрахувати надбавку відповідно до вказаних умов (відсоток від посадового окладу).
4. Додати надбавку до заробітної плати працівника.
5. Вивести суму до виплати.

**Лістинг**

**#include <iostream>**

**using namespace std;**

**float calculateBonus(float base\_salary, float total\_salary, int years\_of\_service) {**

**float bonus\_percentage;**

**if (years\_of\_service >= 3 && years\_of\_service < 5)**

**bonus\_percentage = 0.1;**

**else if (years\_of\_service >= 5 && years\_of\_service < 7)**

**bonus\_percentage = 0.15;**

**else if (years\_of\_service >= 7 && years\_of\_service < 15)**

**bonus\_percentage = 0.20;**

**else**

**bonus\_percentage = 0.25;**

**float bonus\_amount = base\_salary \* bonus\_percentage;**

**float total\_payment = total\_salary + bonus\_amount;**

**return total\_payment;**

**}**

**int main() {**

**float base\_salary, total\_salary;**

**int years\_of\_service;**

**cout << "Введіть розмір посадового окладу (грн.): ";**

**cin >> base\_salary;**

**cout << "Введіть розмір заробітної плати працівника (грн.): ";**

**cin >> total\_salary;**

**cout << "Введіть стаж роботи на підприємстві (років): ";**

**cin >> years\_of\_service;**

**float total\_payment = calculateBonus(base\_salary, total\_salary, years\_of\_service);**

**cout << "Сума до виплати (грн., включно з надбавкою): " << total\_payment << endl;**

**return 0;**

**}**

**Завдання 9.2**

Вхід: температура повітря (у градусах за шкалою Цельсія), зафіксовану о

00:00, 04:00, 8:00, 12:00, 16:00, 20:00 год.

Вихід: середньодобова температура за шкалами Цельсія та Фаренгейта.

**Алгоритм**

1. Ініціалізувати змінну для суми температур за добу.

2. Зчитати температури за кожну з шести годин.

3. Обчислити суму температур за добу.

4. Обчислити середньодобову температуру в градусах Цельсія, поділивши суму на 6.

5. Обчислити середньодобову температуру в градусах Фаренгейта, використовуючи формулу.

6. Вивести середньодобові температури за шкалами Цельсія та Фаренгейта.

**Лістинг**

**#include <iostream>**

**using namespace std;**

**float celsiusToFahrenheit(float celsius) {**

**return 32.0 + (9.0 / 5.0) \* celsius;**

**}**

**int main() {**

**float temperatures[6];**

**float sum\_celsius = 0.0;**

**// Ввід температур за кожну з шести годин**

**for (int i = 0; i < 6; ++i) {**

**cout << "Введіть температуру за " << i \* 4 << ":00 год (градуси Цельсія): ";**

**cin >> temperatures[i];**

**sum\_celsius += temperatures[i];**

**}**

**// Розрахунок середньодобової температури в градусах Цельсія**

**float average\_celsius = sum\_celsius / 6.0;**

**cout << "Середньодобова температура (градуси Цельсія): " << average\_celsius << endl;**

**// Переведення середньодобової температури в градуси Фаренгейта**

**float average\_fahrenheit = celsiusToFahrenheit(average\_celsius);**

**cout << "Середньодобова температура (градуси Фаренгейта): " << average\_fahrenheit << endl;**

**return 0;**

**}**

**Задача 9.3**

Вхід: натуральне число N від 0 до 10008000.

Вихід: якщо біт D11 числа N рівний 0, кількість двійкових нулів у ньому, інакше - кількість двійкових одиниць\*.

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

**Алгоритм**

1. Зчитати вхідне натуральне число N.
2. Перевірити біт D11 числа N. Якщо він дорівнює 0, виконати крок 3, в іншому випадку - крок 4.
3. Обчислити кількість двійкових нулів у числі N.
4. Обчислити кількість двійкових одиниць у числі N.
5. Вивести результат.

**Лістинг**

**#include <iostream>**

**using namespace std;**

**int countZerosOrOnes(int n, bool countZeros) {**

**int count = 0;**

**while (n) {**

**count += (n & 1) == countZeros;**

**n >>= 1;**

**}**

**return count;**

**}**

**int main() {**

**int N;**

**cout << "Введіть натуральне число N (від 0 до 10008000): ";**

**cin >> N;**

**// Перевірка біту D11**

**bool D11\_is\_zero = (N & (1 << 11)) == 0;**

**int result;**

**if (D11\_is\_zero) {**

**// Якщо біт D11 рівний 0, обчислити кількість двійкових нулів у числі N**

**result = countZerosOrOnes(N, true);**

**} else {**

**// Якщо біт D11 не рівний 0, обчислити кількість двійкових одиниць у числі N**

**result = countZerosOrOnes(N, false);**

**}**

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

**return 0;**

**}**

**Задача 9.4**

**Алгоритм**

1. Ініціалізувати змінні.

2. Створити цикл для повторення програми.

3. Очікувати введення користувача та зчитати символ.

4. Обробити введений символ та викликати відповідну функцію або вивести повідомлення про помилку.

5. Повторити кроки 3-4, доки не буде введено символ "q" або "Q".

6. Завершити програму.

**Лістинг**

**#include <iostream>**

**#include <conio.h> // Для отримання символів без очікування натискання клавіші**

**#include <Windows.h> // Для використання функції Beep()**

**using namespace std;**

**// Функція для обчислення s**

**void s\_calculation() {**

**// Реалізація функції s\_calculation**

**cout << "Функція s\_calculation" << endl;**

**}**

**// Функція для задачі 9.1**

**void task\_9\_1() {**

**// Реалізація функції task\_9\_1**

**cout << "Функція task\_9\_1" << endl;**

**}**

**// Функція для задачі 9.2**

**void task\_9\_2() {**

**// Реалізація функції task\_9\_2**

**cout << "Функція task\_9\_2" << endl;**

**}**

**// Функція для задачі 9.3**

**void task\_9\_3() {**

**// Реалізація функції task\_9\_3**

**cout << "Функція task\_9\_3" << endl;**

**}**

**int main() {**

**char input;**

**bool exit = false;**

**do {**

**// Очікуємо введення користувача**

**cout << "Введіть символ (z, r, s, t, q): ";**

**input = \_getch();**

**// Обробка введеного символу**

**switch (input) {**

**case 'z':**

**s\_calculation();**

**break;**

**case 'r':**

**task\_9\_1();**

**break;**

**case 's':**

**task\_9\_2();**

**break;**

**case 't':**

**task\_9\_3();**

**break;**

**case 'q':**

**case 'Q':**

**exit = true;**

**break;**

**default:**

**// Ігноруємо інші символи та видаємо звуковий сигнал про помилку**

**Beep(1000, 500); // Генеруємо звуковий сигнал**

**cout << "Неправильний символ. Спробуйте ще раз." << endl;**

**break;**

**}**

**} while (!exit);**

**cout << "Програма завершила роботу." << endl;**

**return 0;**

**}**