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/](https://github.com/odorenskyi/Dmytro-Parkhomenko-KB18)

ВИКОНАВ

студент академічної групи КН-23 Богданов О.С.

ПЕРЕВІРИВ

ст. викладач кафедри кібербезпеки   
та програмного забезпечення

Дрєєва Г. М.

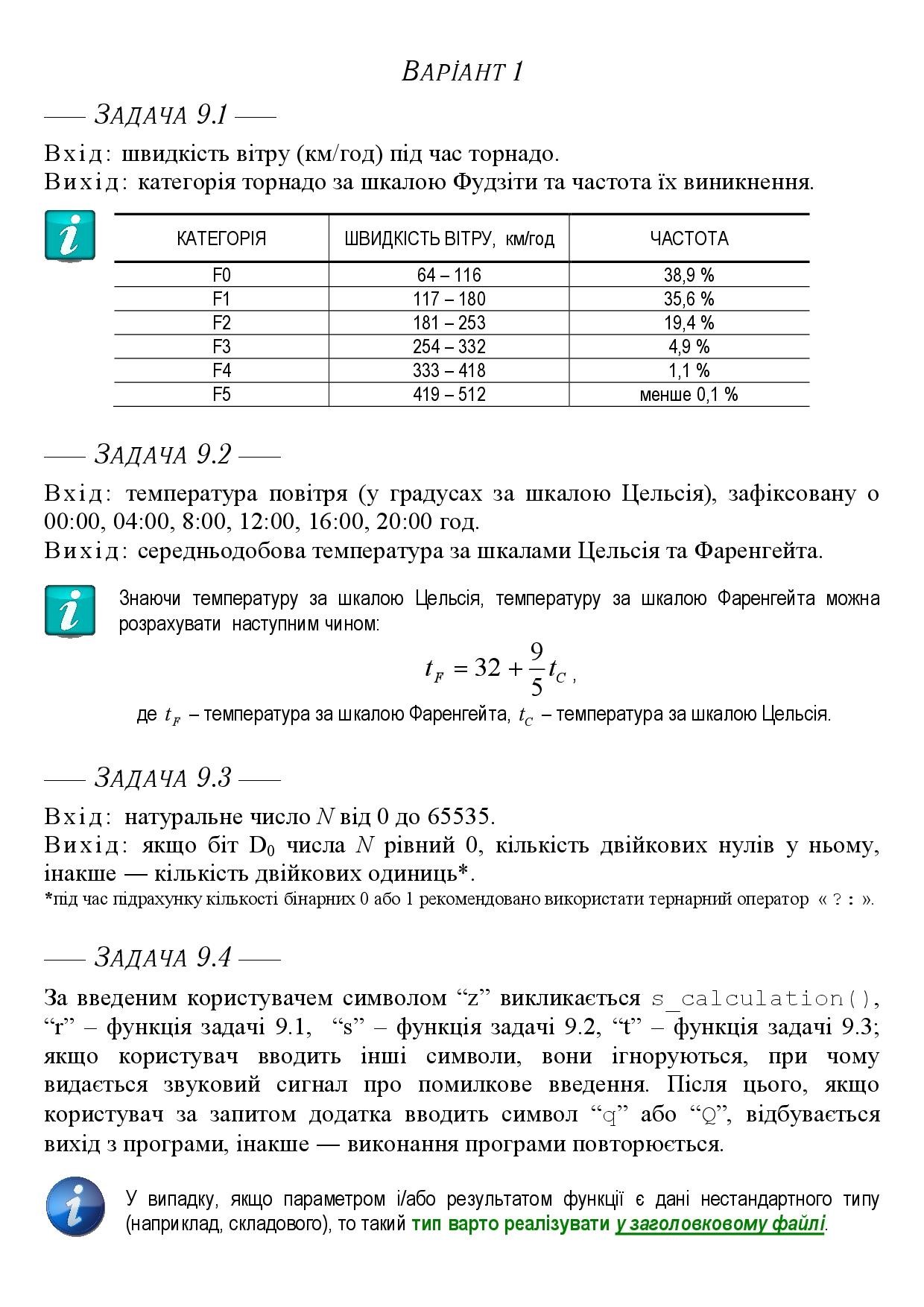
Кропивницький – 2024

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

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

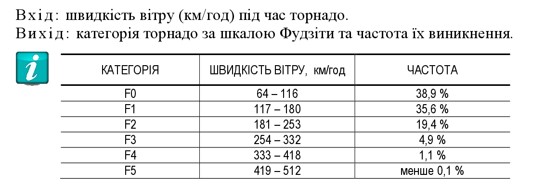
**Завдання:**

* Реалізувати функції розв’язування задач 9.1–9.3 як складових статичної бібліотеки libModulesПрізвище.а (проект ModulesПрізвище, створений під час виконання лабораторної роботи № 8).
* Реалізувати програмне забезпечення розв’язування задачі 9.4 на основі функцій статичної бібліотеки libModulesПрізвище.а



**Варіант 1**

# Завдання 9.1



Вхідні дані: швидкість вітру (км/год) під час торнадо. Вихідні дані: категорія торнадо за шкалою Фудзіти та частота їх виникнення.

**Алгоритм:**

1. Підключаємо статичну бібліотеку модулів ModulesBogdanov.h
2. Запитуємо у користувача speed.
3. Викликаємо функцію printInfoTornado в яку передаємо x, y, z та виводимо результат.

**Лістинг:**

# ModulesBogdanov/ModulesBogdanov.cpp

void printInfoTornado(float speed) { if(64 <= speed && 116 >= speed) {

cout << "Категорія торнадо за шкалою Фудзіти: " << "F0" << endl << "Частота виникнення: 38,9%" << endl;

} else if(117 <= speed && 180 >= speed) {

cout << "Категорія торнадо за шкалою Фудзіти: " << "F1" << endl << "Частота виникнення: 35,6%" << endl;

} else if(181 <= speed && 253 >= speed) {

cout << "Категорія торнадо за шкалою Фудзіти: " << "F2" << endl << "Частота виникнення: 19,4%" << endl;

} else if(254 <= speed && 332 >= speed) {

cout << "Категорія торнадо за шкалою Фудзіти: " << "F3" << endl << "Частота виникнення: 4,9%" << endl;

} else if(333 <= speed && 418 >= speed) {

cout << "Категорія торнадо за шкалою Фудзіти: " << "F4" << endl << "Частота виникнення: 1,1%" << endl;

} else if(419 <= speed && 512 >= speed) {

cout << "Категорія торнадо за шкалою Фудзіти: " << "F5" << endl << "Частота виникнення: меньше 0,1%" << endl;

} else { cout << "Не входить у діапазон" << endl; }

}

# ModulesBogdanov.h

#ifndef MODULESBOGDANOV\_H\_INCLUDED

#define MODULESBOGDANOV\_H\_INCLUDED

float s\_calculation(int x, int y, int z); void printInfoTornado(float speed); void avgTemp(float arr[]);

void countBits(int N);

#endif // MODULESBOGDANOV\_H\_INCLUDED

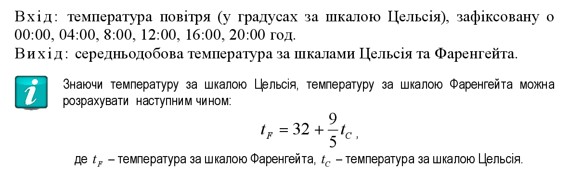
# TestDriver/main.cpp

float speed;

cout << "Введіть швидкість вітру: "; cin >> speed;

printInfoTornado(speed);

# Завдання 9.2



Вхідні дані: темература повітря (у градусах за шкалою Цельсія), зафіксовану о 00:00, 04:00, 08:00, 12:00, 16:00, 20:00 год. Вихідні дані: середньодобова температура за шкалами Цельсія та Фарангейта.

**Алгоритм:**

1. Підключаємо статичну бібліотеку модулів ModulesBogdanov.h
2. Запитуємо у користувача масив temps[6]
3. Викликаємо функцію avgTemp в яку передаємо temps[6] та виводимо результат.

**Лістинг:**

# ModulesBogdanov/ModulesBogdanov.cpp

void avgTemp(float arr[]) {

float sum\_temp = 0;

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

sum\_temp += arr[i];

}

cout << "Цельсії: " << (sum\_temp / 6) << " °C" << endl;

cout << "Фаренгейти: " << ((sum\_temp / 6) \* (9 / 5)) + 32 << " °F" << endl; }

# ModulesBogdanov.h

#ifndef MODULESBOGDANOV\_H\_INCLUDED

#define MODULESBOGDANOV\_H\_INCLUDED

float s\_calculation(int x, int y, int z); void printInfoTornado(float speed); void avgTemp(float arr[]);

void countBits(int N);

#endif // MODULESBOGDANOV\_H\_INCLUDED

# TestDriver/main.cpp

float temps[6];

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

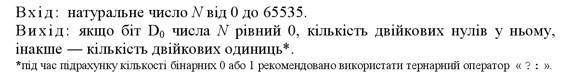
cout << "Введіть температуру: ";

cin >> temp; temps[i] = temp;

}

avgTemp(temps);

# Завдання 9.3



Вхідні дані: натуральне число N від 0 до 65535.

Вихідні дані: якщо біт D0 числа N рівний 0, кількість двійкових нулів у ньому, інакше – кількість двійкових одиниць.

**Алгоритм:**

1. Підключаємо статичну бібліотеку модулів ModulesBogdanov.h 2. Запитуємо у користувача число N.

3. Викликаємо функцію countBits в яку передаємо N та виводимо результат.

**Лістинг:**

# ModulesBogdanov/ModulesBogdanov.cpp

void countBits(int N) {

int counter = 0;

if((N & 1) == 0) {

while (N > 0) { int bit = N % 2;

counter += (bit == 0) ? 1 : 0;

N /= 2;

}

cout << "Кількість двійкових нулів дорівнює: " << counter << endl;

} else {

while (N > 0) { int bit = N % 2;

counter += (bit != 0) ? 1 : 0;

N /= 2;

}

cout << "Кількість двійкових одиниць дорівнює: " << counter << endl;

}

}

# ModulesBogdanov.h

#ifndef MODULESBOGDANOV\_H\_INCLUDED

#define MODULESBOGDANOV\_H\_INCLUDED

float s\_calculation(int x, int y, int z); void printInfoTornado(float speed); void avgTemp(float arr[]); void countBits(int N);

#endif // MODULESBOGDANOV\_H\_INCLUDED

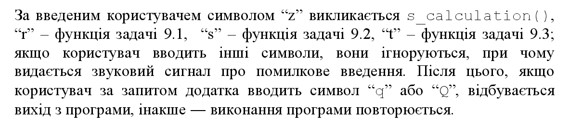
# TestDriver/main.cpp

int N;

cout << "Введіть число: "; cin >> N;

countBits(N);

# Завдання 9.4



Вхідні дані: Символ z, r, s, t, q, Q.

Вихідні дані: Якщо користувач вводить символ “z” викликається s\_calculation(), “r” – функція задачі 9.1, “s” – функція задачі 9.2, “t” – функція задачі 9.3. Якщо користувач вводить інший символ – видати звуковий сигнал про помилкове введення та продовжити цикл. Якщо користувач вводить “q” або “Q” – зробити вихід з програми.

**Алгоритм:**

1. Підключаємо статичну бібліотеку модулів ModulesBogdanov.h
2. Викликаємо нескінченний цикл while.
3. Запитуємо у користувача символ (z, r, s, t, q, Q).
4. Якщо користувач вводить символ “z” викликаємо функцію s\_calculation(), “r” – функція задачі 9.1, “s” – функція задачі 9.2, “t” – функція задачі 9.3. Якщо користувач вводить інший символ – видати звуковий сигнал про помилкове введення та продовжити цикл. Якщо користувач вводить “q” або “Q” – зробити вихід з програми.

**Лістинг:**

# Bogdanov\_task/main.cpp

#include <iostream>

#include <Windows.h>

#include "ModulesBogdanov.h"

using namespace std;

int main() {

system("chcp 65001 & cls");

while (true) { char symbol; cout << "Введіть символ(z, r, s, t, q, Q): "; cin >> symbol;

if (symbol == 'q' || symbol == 'Q') {

break;

}

switch (symbol) { case 'z': int x, y, z;

cout << "Введіть число x: ";

cin >> x; cout << "Введіть число y: ";

cin >> y; cout << "Введіть число z: ";

cin >> z;

cout << s\_calculation(x, y, z) << endl;

break;

case 'r':

float speed;

cout << "Введіть швидкість вітру: ";

cin >> speed;

printInfoTornado(speed);

break;

case 's':

float temps[6];

for(int i = 0; i < 6; i++) { float temp; cout << "Введіть температуру: "; cin >> temp;

temps[i] = temp;

}

avgTemp(temps);

break;

case 't': int N;

cout << "Введіть число: ";

cin >> N;

countBits(N);

break;

default:

Beep(523, 1000);

break;

}

}

return 0;

}

**Висновок:**

У ході лабораторної роботи я успішно вивчив основні принципи модульного програмування та вмію їх застосовувати у практиці. Також я отримав поглиблені знання з операторів мови програмування C++, зокрема арифметичних, логічних та побітових операцій. Засвоївши концепції умов, циклів та вибору, я можу ефективно використовувати їх у своїх програмах для забезпечення правильної роботи та оптимізації коду.