Міністерство освіти і науки України

Центральноукраїнський національний технічний університет

Механіко-технологічний факультет

Кафедра кібербезпеки та програмного забезпечення Дисципліна:

БМТП

**Лабораторна робота №9 В-19**

**Тема:** **« Реалізація Програмних Модулів Розгалуження Та Ітераційних Обчислювальних Процесорів »**

Виконав: ст. гр. КБ-24-1

Іванов Даніїл А.

Перевірив: викладач

Коваленко А.С.

Кропивницький

2025

**ТЕМА: Реалізація Програмних Модулів Розгалуження Та Ітераційних Обчислювальних Процесорів**

**МЕТА: 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.**

**ЗАВДАННЯ:**

Лістинг

#include <iostream>

#include <locale.h>

using namespace std;

int main() {

setlocale(LC\_CTYPE,"ukr");

double waveHeight;

int grade = -1;

cout << "Введіть висоту хвилі у футах (FT, не цілі числа писати через крапку): ";

cin >> waveHeight;

if (waveHeight == 0)

grade = 0;

else if (waveHeight > 0 && waveHeight <= 0.3)

grade = 1;

else if (waveHeight > 0.3 && waveHeight <= 1.3)

grade = 2;

else if (waveHeight > 1.3 && waveHeight <= 4)

grade = 3;

else if (waveHeight > 4 && waveHeight <= 8)

grade = 4;

else if (waveHeight > 8 && waveHeight <= 13)

grade = 5;

else if (waveHeight > 13 && waveHeight <= 20)

grade = 6;

else if (waveHeight > 20 && waveHeight <= 30)

grade = 7;

else if (waveHeight > 30 && waveHeight <= 45)

grade = 8;

else if (waveHeight > 45)

grade = 9;

switch (grade) {

case 0: cout << "Grade: 0 - CALM-GLASSY";

break;

case 1: cout << "Grade: 1 - CALM-RIPPLED";

break;

case 2: cout << "Grade: 2 - SMOOTH-WAVELET";

break;

case 3: cout << "Grade: 3 - SLIGHT";

break;

case 4: cout << "Grade: 4 - MODERATE";

break;

case 5: cout << "Grade: 5 - ROUGH";

break;

case 6: cout << "Grade: 6 - VERY ROUGH";

break;

case 7: cout << "Grade: 7 - HIGH";

break;

case 8: cout << "Grade: 8 - VERY HIGH";

break;

case 9: cout << "Grade: 9 - PHENOMENAL";

break;

default: cout << "Некоректна висота хвилі.";

break;

}

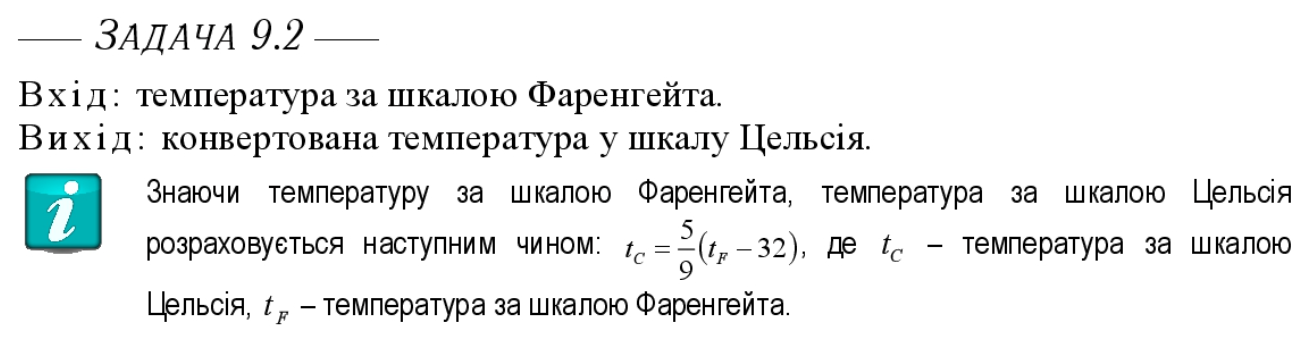
cout << endl;

return 0;

}

Псевдокод

1. Встановити українську локалізацію
2. Оголосити змінну waveHeight (тип double)
3. Оголосити змінну grade = -1 (тип int)
4. Вивести на екран: "Введіть висоту хвилі у футах (FT, не цілі числа писати через крапку):"
5. Зчитати значення waveHeight
6. Якщо waveHeight == 0: grade = 0 Інакше якщо waveHeight > 0 і waveHeight ≤ 0.3: grade = 1 Інакше якщо waveHeight > 0.3 і waveHeight ≤ 1.3: grade = 2 Інакше якщо waveHeight > 1.3 і waveHeight ≤ 4: grade = 3 Інакше якщо waveHeight > 4 і waveHeight ≤ 8: grade = 4 Інакше якщо waveHeight > 8 і waveHeight ≤ 13: grade = 5 Інакше якщо waveHeight > 13 і waveHeight ≤ 20: grade = 6 Інакше якщо waveHeight > 20 і waveHeight ≤ 30: grade = 7 Інакше якщо waveHeight > 30 і waveHeight ≤ 45: grade = 8 Інакше якщо waveHeight > 45: grade = 9
7. Виконати вибір залежно від значення grade:
   1. Якщо grade == 0: Вивести "Grade: 0 - CALM-GLASSY"
   2. Якщо grade == 1: Вивести "Grade: 1 - CALM-RIPPLED"
   3. Якщо grade == 2: Вивести "Grade: 2 - SMOOTH-WAVELET"
   4. Якщо grade == 3: Вивести "Grade: 3 - SLIGHT"
   5. Якщо grade == 4: Вивести "Grade: 4 - MODERATE"
   6. Якщо grade == 5: Вивести "Grade: 5 - ROUGH"
   7. Якщо grade == 6: Вивести "Grade: 6 - VERY ROUGH"
   8. Якщо grade == 7: Вивести "Grade: 7 - HIGH"
   9. Якщо grade == 8: Вивести "Grade: 8 - VERY HIGH"
   10. Якщо grade == 9: Вивести "Grade: 9 - PHENOMENAL"
   11. Інакше: Вивести "Некоректна висота хвилі."
8. Завершити програму

Лістинг

#include <iostream>

#include <locale.h>

using namespace std;

int main() {

setlocale(LC\_CTYPE,"ukr");

double tf, tc;

// Ввід температури за шкалою Фаренгейта

cout << "Введіть температуру у Фаренгейтах: ";

cin >> tf;

// Обчислення температури у Цельсіях

tc = 5.0 / 9.0 \* (tf - 32);

// Вивід результату

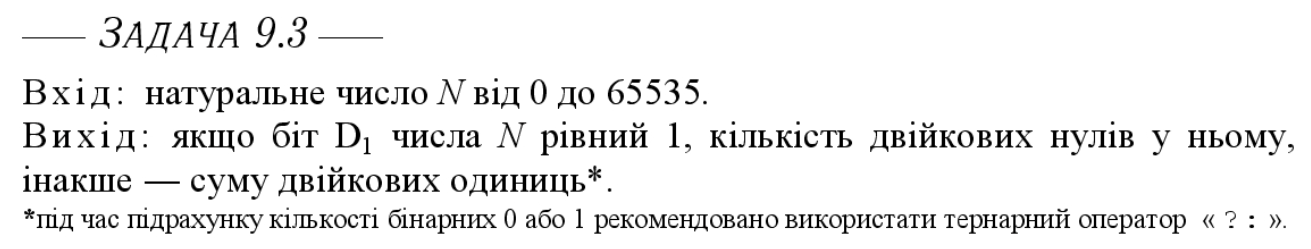
cout << "Температура у Цельсіях: " << tc << endl;

return 0;

}

Псевдокод

1. Встановити українську локалізацію
2. Оголосити змінні:
   1. tf (температура у Фаренгейтах, тип double)
   2. tc (температура у Цельсіях, тип double)
3. Вивести на екран: "Введіть температуру у Фаренгейтах:"
4. Зчитати значення tf
5. Обчислити температуру в Цельсіях: tc = (5 / 9) \* (tf - 32)
6. Вивести результат: "Температура у Цельсіях: " + tc
7. Завершити програму

Лістинг

#include <iostream>

#include <bitset>

#include <locale.h>

using namespace std;

int main() {

setlocale(LC\_CTYPE,"ukr");

unsigned int N;

cout << "Введіть натуральне число N (від 0 до 65535): ";

cin >> N;

if (N > 65535) {

cout << "Число повинно бути в межах від 0 до 65535." << endl;

return 1;

}

// Перевіряємо біт D1 (тобто другий справа біт, індексація з 0)

bool D1 = (N >> 1) & 1;

// Конвертуємо N в двійковий вигляд для підрахунку нулів та одиниць

bitset<16> binary(N);

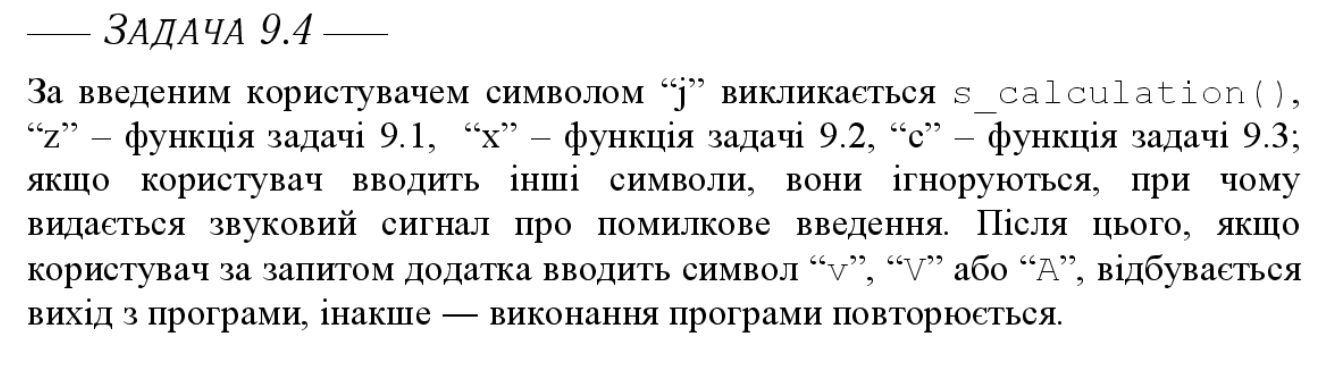
int result = D1 ? (16 - binary.count()) : binary.count(); // тернарний оператор

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

return 0;

}

1. Псевдокод  
   Встановити українську локалізацію
2. Оголосити змінну N (тип unsigned int)
3. Вивести повідомлення: "Введіть натуральне число N (від 0 до 65535):"
4. Зчитати значення N
5. Якщо N > 65535: Вивести: "Число повинно бути в межах від 0 до 65535." Завершити програму з кодом 1
6. Визначити значення біта D1:
   1. D1 = значення другого справа біта в числі N
   2. Для цього виконуємо зсув N вправо на 1 біт і логічне "І" з 1: D1 = (N >> 1) & 1
7. Перетворити число N у 16-бітове двійкове представлення (bitset з 16 біт)
8. Якщо D1 == 1: Порахувати кількість нулів: result = 16 - кількість одиниць Інакше: Порахувати кількість одиниць: result = кількість одиниць
9. Вивести результат: "Результат: " + result
10. Завершити програму

Лістинг

#include <iostream>

#include <string>

#include <bitset>

#include <locale.h>

using namespace std;

// Завдання 9.1

string seaState(double height) {

if (height == 0) return "CALM-GLASSY";

else if (height <= 0.1) return "CALM-RIPPLE";

else if (height <= 0.5) return "SMOOTH-WAVELET";

else if (height <= 1.25) return "SLIGHT";

else if (height <= 2.5) return "MODERATE";

else if (height <= 4) return "ROUGH";

else if (height <= 6) return "VERY ROUGH";

else if (height <= 9) return "HIGH";

else if (height <= 14) return "VERY HIGH";

else return "PHENOMENAL";

}

// Завдання 9.2

double fahrenheitToCelsius(double tf) {

return (5.0 / 9.0) \* (tf - 32);

}

// Завдання 9.3

int countBits(int N, bool countZero) {

bitset<16> b(N);

int count = 0;

for (int i = 0; i < b.size(); i++) {

if (countZero && b[i] == 0) count++;

else if (!countZero && b[i] == 1) count++;

}

return count;

}

// Завдання 9.4

void s\_calculation(char symbol) {

setlocale(LC\_CTYPE,"ukr");

if (symbol == 'j') {

double h;

cout << "Введіть висоту хвиль (м): ";

cin >> h;

cout << "Стан моря: " << seaState(h) << endl;

} else if (symbol == 'x') {

double tf;

cout << "Введіть температуру за Фаренгейтом: ";

cin >> tf;

cout << "Температура за Цельсієм: " << fahrenheitToCelsius(tf) << endl;

} else if (symbol == 'q') {

int N;

cout << "Введіть натуральне число N (до 65535): ";

cin >> N;

cout << "Виберіть варіант: 1 — рахуємо нулі, інше — одиниці: ";

int D;

cin >> D;

cout << "Кількість бітів: " << countBits(N, D == 1) << endl;

} else {

cout << "Невідома команда!" << endl;

}

}

int main() {

setlocale(LC\_CTYPE,"ukr");

char cmd;

cout << "Введіть команду ('j', 'x', 'q'): ";

cin >> cmd;

s\_calculation(cmd);

return 0;

}

Псевдокод

Встановити українську локалізацію

Вивести повідомлення: "Введіть команду ('j', 'x', 'q'):"

Зчитати символ команди

Викликати функцію s\_calculation з переданою командою

Якщо symbol == 'j':

Вивести: "Введіть висоту хвиль (м):"

Зчитати висоту

Вивести результат seaState(висота)

Інакше якщо symbol == 'x':

Вивести: "Введіть температуру за Фаренгейтом:"

Зчитати температуру

Вивести результат fahrenheitToCelsius(температура)

Інакше якщо symbol == 'q':

Вивести: "Введіть натуральне число N (до 65535):"

Зчитати N

Вивести: "Виберіть варіант: 1 — рахуємо нулі, інше — одиниці:"

Зчитати D

Якщо D == 1:

countZero = true

Інакше:

countZero = false

Вивести результат countBits(N, countZero)

Інакше:

Вивести: "Невідома команда!"

Аргументи

* Використання #include для підключення бібліотек
* Робота з простором імен std
* Основи структури main() функції
* Оголошення змінних різного типу: int, double, bool, unsigned int
* Використання setlocale() для української мови в консолі
* Зчитування значень через cin
* Виведення тексту на екран через cout
* Формулювання зрозумілих інтерфейсів для користувача
* Розуміння if, else if, else
* Перевірка значень на межі (наприклад, перевірка N <= 65535)
* Обробка помилкових ввідних даних
* Використання тернарного оператора ? : для скорочених умов
* Поняття **бітового зсуву** (>>)
* Використання побітового "І" (&) для перевірки окремих бітів
* Робота з bitset<16> для представлення числа в двійковому вигляді
* Підрахунок кількості одиниць через .count()
* Логіка аналізу бітових структур
* Перетворення температури з Фаренгейта у Цельсій
* Реалізація формули: C = (5/9)\*(F - 32)
* Обчислення висоти хвиль та класифікація їх за шкалою
* Використання діапазонів у порівняннях (наприклад 0 < x <= 0.3)
* Заміна довгих if-else на switch для читабельності
* Створення варіантів вибору залежно від змінної
* Забезпечення дефолтного випадку (default)
* Поділ програми на окремі логічні блоки
* Створення функцій, які повертають значення
* Створення функцій без повернення (void)
* Робота з параметрами функцій
* Передача даних у функцію
* Перетворення коду у зрозумілий **алгоритмічний опис**
* Структурування дій крок за кроком
* Покращення навичок аналізу алгоритмів
* Використання псевдокоду як підготовка до написання реального коду
* Форматування коду для читабельності
* Коментування коду
* Вибір зрозумілих імен змінних
* Пояснення коду для інших (і для себе у майбутньому)
* Уміння будувати умови й перевірки
* Планування дій до написання коду
* Відстеження помилок в логіці програми
* Побудова алгоритму на основі задачі
* Розробка простих консолних застосунків
* Написання багатофункціональної програми з меню
* Навчання на прикладах: хвилі, температура, біти
* Використання локалізації в програмах (українська мова)
* Розуміння циклу: ввід → обробка → вивід
* Впевненість у роботі з базовим C++
* Навички писати зрозумілий, логічно структурований код
* Усвідомлення, як можна розширити програму (наприклад, додати нові функції)
* Розуміння, як з простих блоків складається повноцінна програма