Міністерство освіти і науки України

Чернівецький національний університет імені Юрія Федьковича

ВСП «Фаховий коледж ЧНУ імені Юрія Федьковича»

Звіт

про виконання лабораторної роботи №2

«Побітові операції»

з дисципліни

«Об’єктно-орієнтоване програмування»

Виконала: студентка 31-КН групи

К.В. Ковальська

Перевірив: канд. фіз.-мат. Наук

доцент Лазорик В.В.

Оцінка:

Дата захисту:

Чернівці 2023 р.

**Завдання 1.** Обчислення виразів з використанням побітових операцій. Задано цілі числа a, b, c та d. Обчислити вираз без використання операцій множення та ділення(замінивши на їх операцій зсувів).



#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "ukr");

int a, b, c, d;

cout << "Введіть значення a: ";

cin >> a;

cout << "Введіть значення b: ";

cin >> b;

cout << "Введіть значення c: ";

cin >> c;

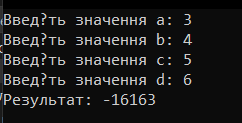
cout << "Введіть значення d: ";

cin >> d;

int result = (b << 10) + ((d << 4) + (a << 3)) - ((c << 12) - c) + (d << 4);

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

return 0;}



**Завдання 2.** 10. Задано текст, який складається з 4 рядків до 16 символів у рядку. Доповнити пробілами рядки до 16 символів. Шифрувати тексти таким чином, щоб кожний символ тексту записувався у два байта. Два байта мають таку структуру: у бітах 0-3 молодша частина ASCII - коду символу (4 біти), у бітах 4-5 знаходиться номер рядка букви (2 біти), у бітах 6-9 позиція букви у рядку (4 біти), у бітах 10-13 старша частина ASCII - коду символу (4 біти), у бітах 6-13 ASCII - код символу (8 біт), 14 біт – біт парності полів 4-9 (1 біт) 15 біт - біт парності коду символу (1 біт)

#include <iostream>

#include <string>

std::string padTextTo16Chars(const std::string& text) {

std::string paddedText = text;

while (paddedText.length() < 16) {

paddedText += ' ';

}

return paddedText;

}

std::string encryptText(const std::string& text) {

std::string encryptedText;

for (char c : text) {

char lowerPart = c & 0x0F;

char upperPart = (c >> 4) & 0x0F;

char rowPos = ((c >> 2) & 0x0F) | (((c >> 6) & 0x03) << 4);

char parity = (c & 1) ^ ((c >> 1) & 1);

char byte1 = lowerPart | (rowPos << 4) | (parity << 6);

char byte2 = upperPart | (parity << 4);

encryptedText += byte1;

encryptedText += byte2;

}

return encryptedText;

}

int main() {

setlocale(LC\_ALL, "ukr");

std::string text;

std::cout << "Введіть текст (до 4 рядків по 16 символів у кожному):\n";

for (int i = 0; i < 4; i++) {

std::string line;

std::getline(std::cin, line);

text += padTextTo16Chars(line);

}

std::string encryptedText = encryptText(text);

std::cout << "Зашифрований текст у вигляді шістнадцяткових байт:\n";

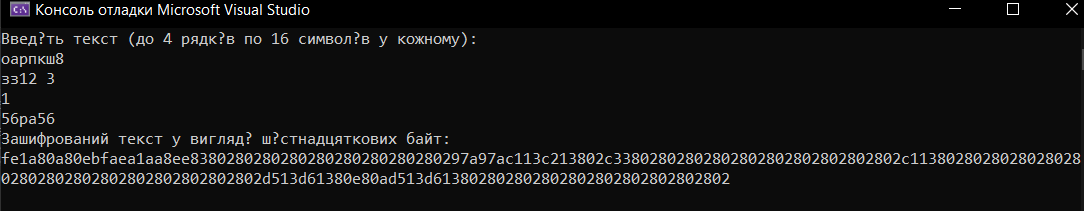
for (char c : encryptedText) {

std::cout << std::hex << static\_cast<int>(static\_cast<unsigned char>(c));

}

std::cout << std::dec << std::endl;

return 0;}



Завдання 3. Реалізувати завдання 2 з використанням структур з бітовими полями та об’єднаннями

#include <iostream>

#include <string>

// структура, що представляє біти символу

struct CharacterBits {

unsigned char lowerPart : 4; // молодша частина ASCII-коду (4 біти)

unsigned char row : 2; // номер рядка букви (2 біти)

unsigned char position : 4; // позиція букви у рядку (4 біти)

unsigned char upperPart : 4; // старша частина ASCII-коду (4 біти)

unsigned char codeParity : 1; // біт парності полів 4-9 (1 біт)

unsigned char charParity : 1; // біт парності коду символу (1 біт)

};

// об'єднання для зберігання двох байт

union TwoBytes {

unsigned short value;

CharacterBits bits;

};

std::string padTextTo16Chars(const std::string& text) {

std::string paddedText = text;

while (paddedText.length() < 16) {

paddedText += ' ';

}

return paddedText;

}

std::string encryptText(const std::string& text) {

std::string encryptedText;

for (char c : text) {

TwoBytes encodedChar;

encodedChar.bits.lowerPart = c & 0x0F;

encodedChar.bits.upperPart = (c >> 4) & 0x0F;

encodedChar.bits.row = 0; // замініть на відповідний номер рядка

encodedChar.bits.position = 0; // замініть на відповідну позицію

encodedChar.bits.codeParity = 0; // розрахує біт парності для полів 4-9

encodedChar.bits.charParity = 0; // розрахує біт парності для коду символу

encryptedText += static\_cast<char>(encodedChar.value & 0xFF);

encryptedText += static\_cast<char>((encodedChar.value >> 8) & 0xFF);

}

return encryptedText;

}

int main() {

setlocale(LC\_ALL, "ukr");

std::string text;

std::cout << "Введіть текст (до 4 рядків по 16 символів у кожному):\n";

for (int i = 0; i < 4; i++) {

std::string line;

std::getline(std::cin, line);

text += padTextTo16Chars(line);

}

std::string encryptedText = encryptText(text);

std::cout << "Зашифрований текст у вигляді шістнадцяткових байт:\n";

for (char c : encryptedText) {

std::cout << std::hex << static\_cast<int>(static\_cast<unsigned char>(c));

}

std::cout << std::dec << std::endl;

return 0;}

