

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів
Кафедра систем управління літальних апаратів

Лабораторна робота № 9

з дисципліни «Алгоритмізація та програмування»

Тема: "Реалізація алгоритмів сортування та робота з файлами на мові C ++"

ХАІ.301. 174. 319. 12 ЛР

Виконав студент гр. 319

Віталій НОВИКОВ

(підпис, дата)

(П.І.Б.)

Перевірив

к.т.н., доц. Олена ГАВРИЛЕНКО

(підпис, дата)

(П.І.Б.)

МЕТА РОБОТИ

Вивчити теоретичний матеріал з основ роботи з низькорівневими рядками на C++ і документацію до класу string, а також алгоритми пошуку в рядку, а також реалізувати обробку рядків на C++ в середовищі Visual Studio.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Виконати шифрування тексту зі зміщенням на 3 та розшифрувати текст назад.

Завдання 2. Підрахувати кількості слів у рядку та зробити реверс рядка.

ВИКОНАННЯ РОБОТИ

Завдання 1. Виконуємо шифрування тексту зі зміщенням на 3 та розшифровуємо текст назад.

Вирішення задачі: task1().

Вхідні дані:

- text - string.

Вихідні дані:

- encrypted - string.
- decrypted - string.

Алгоритм вирішення представлено в додатку Б, рис. Б.1(стор.6).

Завдання 2. Підрахунок кількості слів у рядку та реверс рядка.

Вирішення задачі: task2().

Вхідні дані:

- text - string.

Вихідні дані:

- wordCount - int.
- reversed - string.

Алгоритм вирішення представлено в додатку Б, рис. Б.2(стор.7).

Лістинг коду вирішення задач наведено в додатку А (стор. 3).

Екран роботи програми показаний в додатку Б, рис. Б.3(стор.8).

ВИСНОВКИ

У ході виконання лабораторної роботи була розроблена програма на C++, яка складається з двох задач. Загалом, робота поглибила знання з програмування, покращила навички алгоритмічного мислення та навчила ефективно взаємодіяти з користувачем.

ДОДАТОК А

Лістинг коду програми

main.cpp

```
// Головний файл програми
// Використовується для демонстрації роботи всіх задач

#include <iostream>          // Підключення стандартної бібліотеки для
введення/виведення
#include "task1_encryptor.h" // Підключення заголовочного файлу шифрування
#include "task2_stringutils.h" // Підключення заголовочного файлу роботи з
рядками

using namespace std;

int main() {
    int choice;
    string text;

    while (true) {
        // Меню вибору завдання
        cout << "\nМеню вибору:" << endl;
        cout << "1. Шифрування тексту зі зміщенням на 3" << endl;
        cout << "2. Підрахунок кількості слів у рядку" << endl;
        cout << "0. Вихід" << endl;
        cout << "Введіть вибір (1, 2 або 0): ";
        cin >> choice;
        cin.ignore(); // Очищення буфера введення

        if (choice == 0) {
            cout << "Вихід з програми..." << endl;
            break; // Вихід з програми
        }

        if (choice == 1) {
            // Введення користувачем рядка для шифрування
            string text;
            cout << "Введіть текст для шифрування: ";
            getline(cin, text);

            // Виконуємо шифрування тексту зі зміщенням на 3
            string encrypted = encryptCaesar(text, 3);
            cout << "Зашифрований текст: " << encrypted << endl;

            // Розшифровуємо текст назад
            string decrypted = decryptCaesar(encrypted, 3);
            cout << "Розшифрований текст: " << decrypted << endl;
        }
        else if (choice == 2) {
            // Введення користувачем речення для роботи з рядками
            cout << "\nВведіть речення: ";
            getline(cin, text);

            // Підрахунок кількості слів у рядку
            int wordCount = countWords(text);
            cout << "Кількість слів у реченні: " << wordCount << endl;

            // Реверс рядка
            string reversed = reverseString(text);
            cout << "Реверсований рядок: " << reversed << endl;
        }
        else {
            cout << "Невірний вибір! Спробуйте ще раз." << endl;
        }
    }
}
```

```

    }
    return 0; // Завершення програми
}

```

task1_encryptor.h

```

// Заголовковий файл для функцій шифрування Цезаря

#ifndef TASK1_ENCRYPTOR_H
#define TASK1_ENCRYPTOR_H

#include <string> // Підключення бібліотеки для роботи з рядками

// Функція для шифрування рядка методом Цезаря
// Приймає: вихідний рядок та кількість зміщень
// Повертає: зашифрований рядок
std::string encryptCaesar(const std::string& text, int shift);

// Функція для розшифрування рядка методом Цезаря
// Приймає: зашифрований рядок та кількість зміщень
// Повертає: розшифрований рядок
std::string decryptCaesar(const std::string& text, int shift);

#endif // TASK1_ENCRYPTOR_H

```

task1_encryptor.cpp

```

// Файл реалізації функцій шифрування Цезаря

#include "task1_encryptor.h" // Підключаємо заголовковий файл

// Функція для шифрування тексту методом Цезаря
std::string encryptCaesar(const std::string& text, int shift) {
    std::string result = text; // Копіюємо вхідний текст

    // Проходимо по кожному символу рядка
    for (char& c : result) {
        // Перевіряємо, чи є символ літерою латинського алфавіту
        if (isalpha(c)) {
            char base = isupper(c) ? 'A' : 'a'; // Визначаємо базовий символ (А
чи а)
            c = (c - base + shift) % 26 + base; // Виконуємо зміщення
        }
    }
    return result; // Повертаємо зашифрований рядок
}

// Функція для розшифрування тексту методом Цезаря
std::string decryptCaesar(const std::string& text, int shift) {
    return encryptCaesar(text, 26 - shift); // Використовуємо зворотне зміщення
}

```

task2_stringutils.h

```

// Заголовковий файл для утиліт роботи з рядками

#ifndef TASK2_STRINGUTILS_H
#define TASK2_STRINGUTILS_H

#include <string> // Підключення бібліотеки для роботи з рядками

// Функція підрахунку кількості слів у рядку
// Приймає: вихідний рядок
// Повертає: кількість слів
int countWords(const std::string& text);

// Функція реверсу рядка
// Приймає: вихідний рядок

```

```
// Повертає: реверсований рядок
std::string reverseString(const std::string& text);
```

```
#endif // TASK2_STRINGUTILS_H
```

task2_stringutils.cpp

```
// Файл реалізації утиліт роботи з рядками
```

```
#include "task2_stringutils.h" // Підключаємо заголовковий файл
#include <sstream> // Підключаємо бібліотеку для роботи з потоками
```

```
// Функція підрахунку кількості слів у рядку
```

```
int countWords(const std::string& text) {
    std::istringstream stream(text); // Створюємо потік для розбиття рядка на
    слова
```

```
    int count = 0; // Лічильник слів
    std::string word; // Змінна для зберігання окремого слова
```

```
    // Читаємо слова з потоку
```

```
    while (stream >> word) {
        count++; // Збільшуємо лічильник слів
```

```
    }
    return count; // Повертаємо загальну кількість слів
```

```
}
```

```
// Функція реверсу рядка
```

```
std::string reverseString(const std::string& text) {
```

```
    return std::string(text.rbegin(), text.rend()); // Використовуємо реверсний
    ітератор
```

```
}
```

ДОДАТОК Б

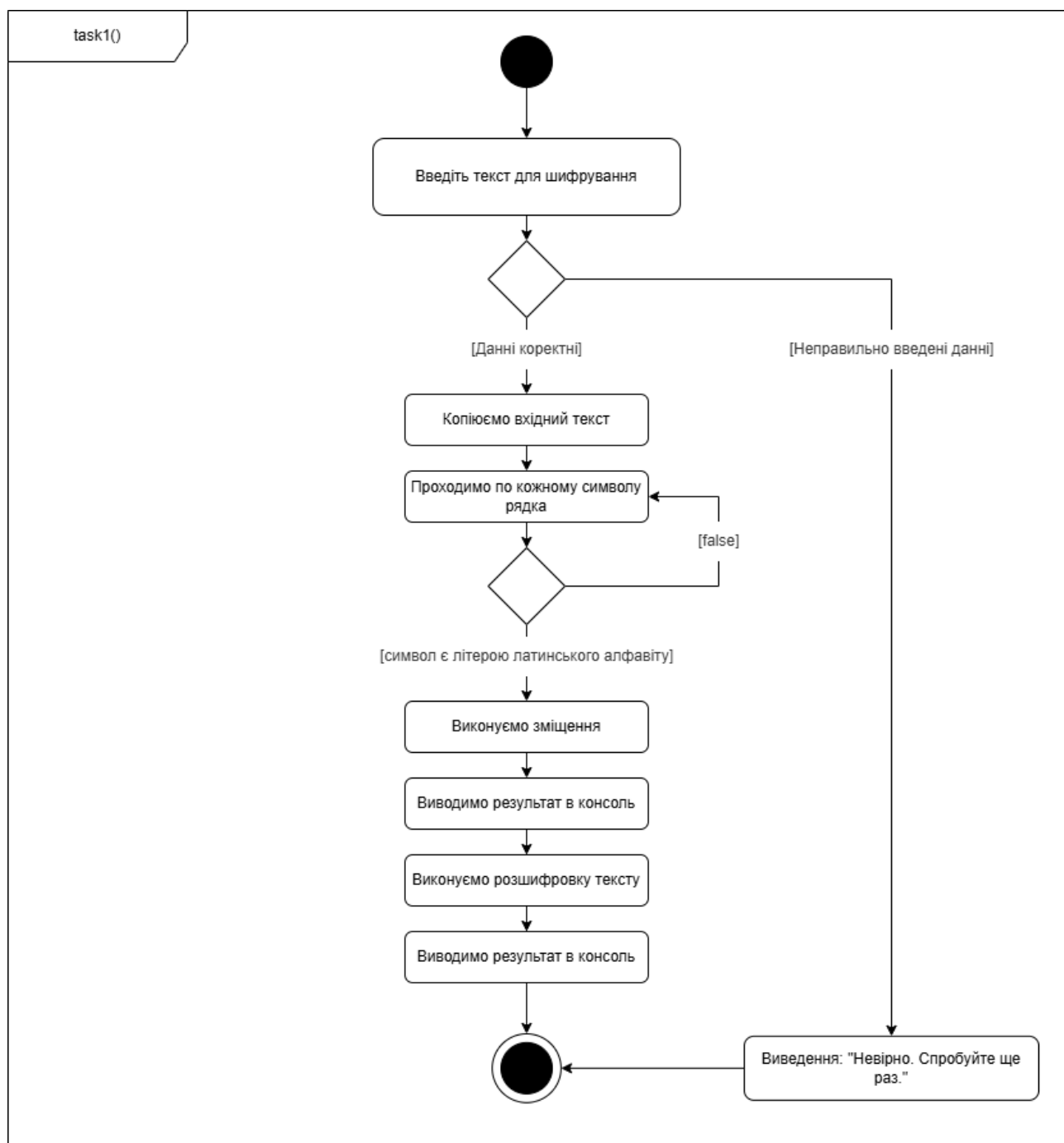


Рисунок Б.1 – діаграма активності task1().

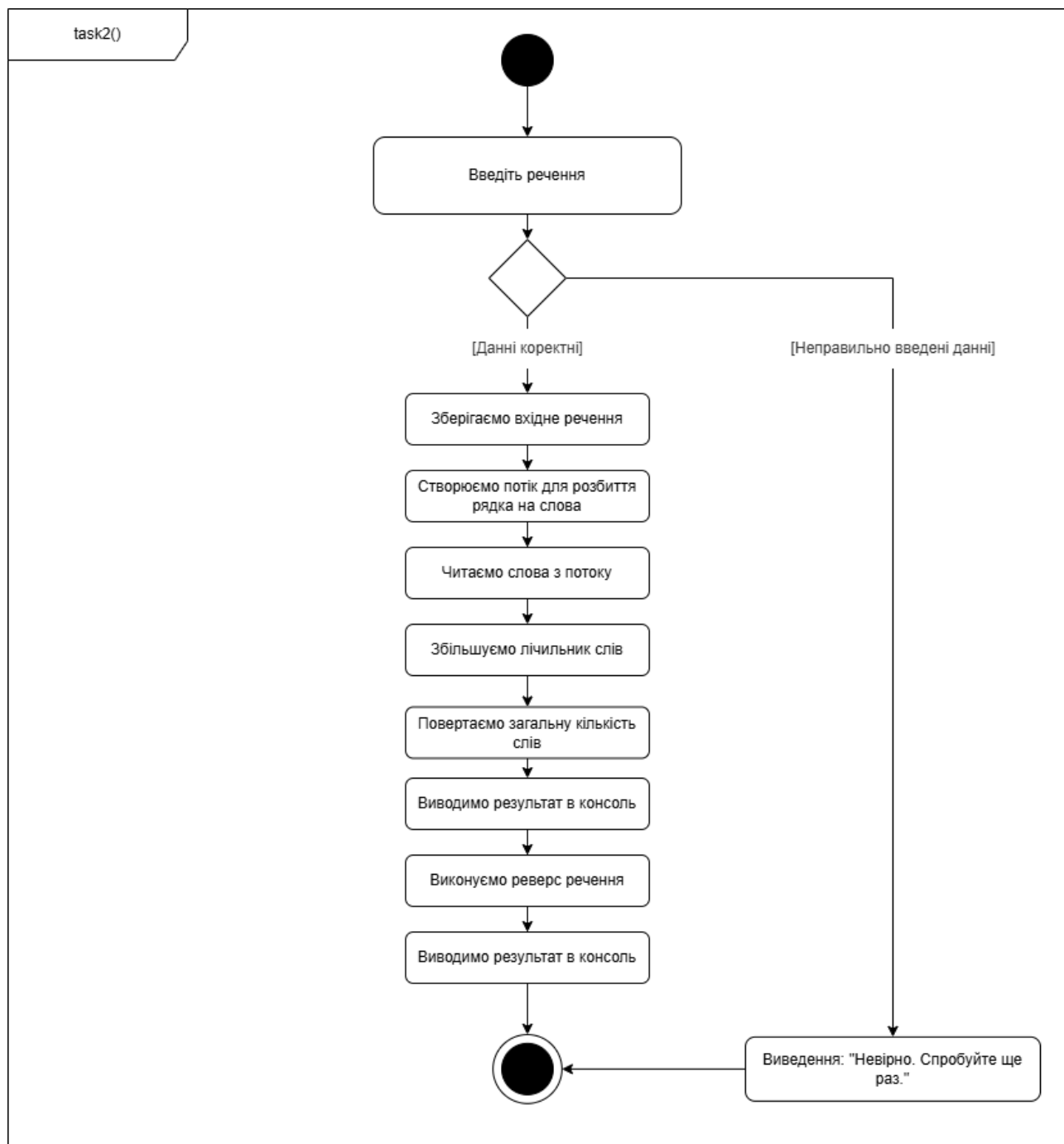


Рисунок Б.2 – Діаграма активності task2().

```
Меню вибору:
1. Шифрування тексту зі зміщенням на 3
2. Підрахунок кількості слів у рядку
0. Вихід
Введіть вибір (1, 2 або 0): 1
Введіть текст для шифрування: abc kkk
Зашифрований текст: def nnn
Розшифрований текст: abc kkk

Меню вибору:
1. Шифрування тексту зі зміщенням на 3
2. Підрахунок кількості слів у рядку
0. Вихід
Введіть вибір (1, 2 або 0): 2

Введіть речення: df wlmow fkv dkmge, rk!
Кількість слів у реченні: 5
Реверсований рядок: !kr ,egmkd vkf womlw fd

Меню вибору:
1. Шифрування тексту зі зміщенням на 3
2. Підрахунок кількості слів у рядку
0. Вихід
Введіть вибір (1, 2 або 0): 0
Вихід з програми...
```

Рисунок Б.3 – Екран роботи програми.