

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів
Кафедра систем управління літальних апаратів

Лабораторна робота № 9

з дисципліни «Алгоритмізація та програмування»
на тему «Робота з рядками на C ++»

XAI.301.173.310.02 ЛР

Виконав студент гр. _____ 310 _____

12.05.2024 _____ Софія ПОЛЯКОВА _____
(підпис, дата) (П.І.Б.)

Перевірив

_____ к.т.н., доц. Олена ГАВРИЛЕНКО
(підпис, дата) (П.І.Б.)

2024

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.

- А. Вивчити по документації метод стандартного класу string відповідно до варіанту.
- В. Визначити функцію, що виконує ті ж дії, що і вивчений метод класу string. Вихідний рядок передати першим параметром (масив символів). Для реалізації методу не використовувати функції обробки рядків зі стандартних бібліотек.
- С. Викликати свій метод і метод string.

Завдання 45. `int compare (size_t pos, size_t len, const char* s) const;`

Завдання 2.

- А. Описати функцію, що обробляє рядок відповідно до завдання. Для реалізації можна використовувати функції обробки рядків зі стандартних бібліотек
- В. Описати функцію, яка перевіряє, чи задовольняє рядок умовам завдання.
- С. Створити вихідний текстовий файл, що містить не менше 10 різних рядків.
- Д. Використовуючи функції 2.А і 2.В, обробити рядок текстовий файл рядок за рядком. Додаткові дані ввести з консолі.
- Е. Отриманий результат записати у вихідний файл.

String13. Дано рядок. Підрахувати кількість цифр в ньому.

ВИКОНАННЯ РОБОТИ

Завдання 1.

Вирішення задачі Завдання 1-45

Вхідні дані (ім'я, опис, тип, обмеження):

- 1) str1 — фраза “wonderful world”;
- 2) str2 — фраза “wonderful world”.

Вихідні дані (ім'я, опис, тип):

- 1) comp_res — число, наскільки не рівне значення, ціле число.

Алгоритм вирішення показано на рис. 1

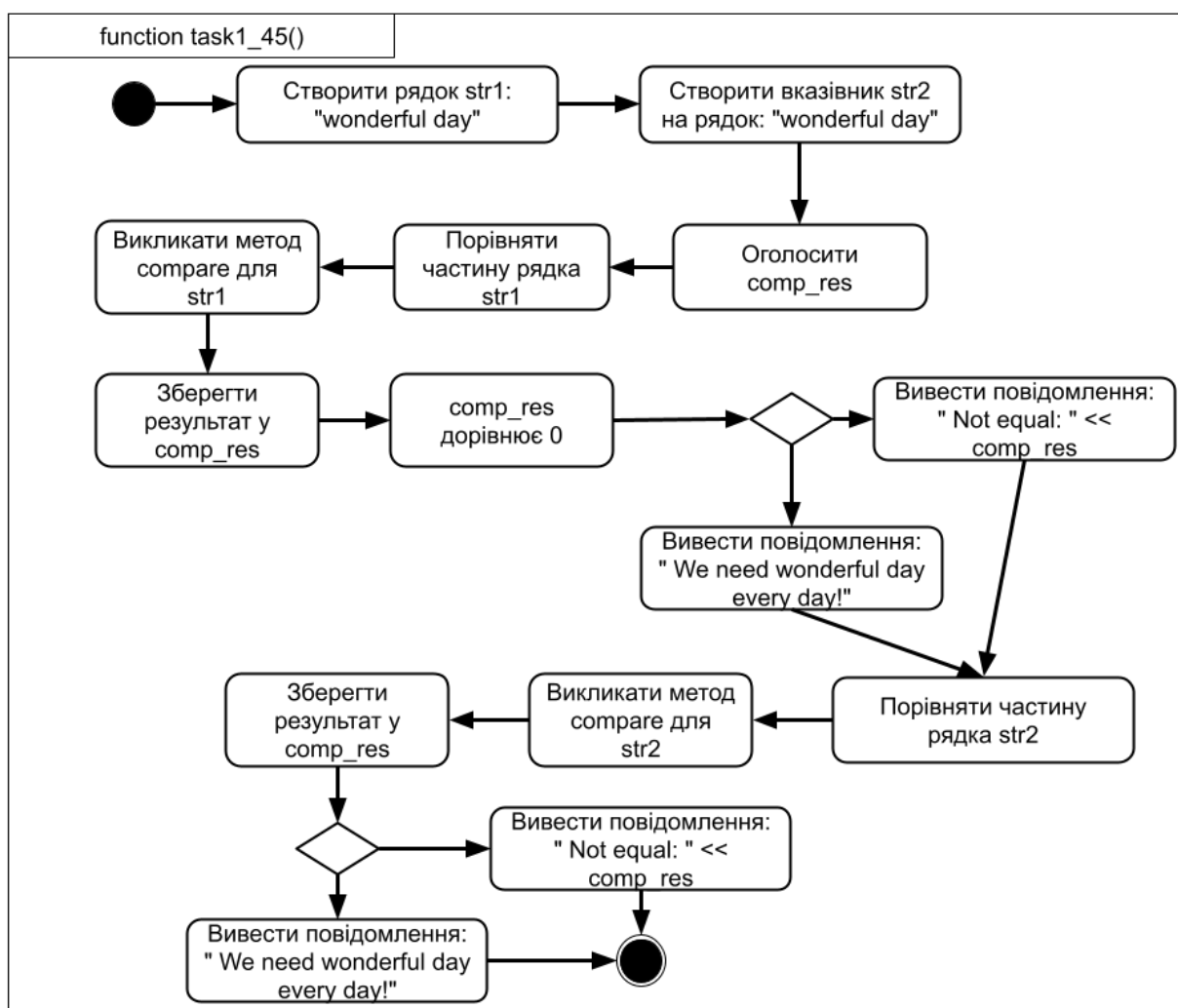


Рисунок 1 — Алгоритм вирішення Завдання 1-45

Лістинг коду вирішення задачі Завдання 1-45 наведено в дод. А (стор. 6).

Екран роботи програми показаний на рис. Б.1.

Завдання 2.

Вирішення задачі String13.

Вхідні дані (ім'я, опис, тип, обмеження):

threshold — порогове значення, ціле число.

Вихідні дані (ім'я, опис, тип):

line — рядок чисел, дійсний тип.

Алгоритм вирішення показано на рис. 2

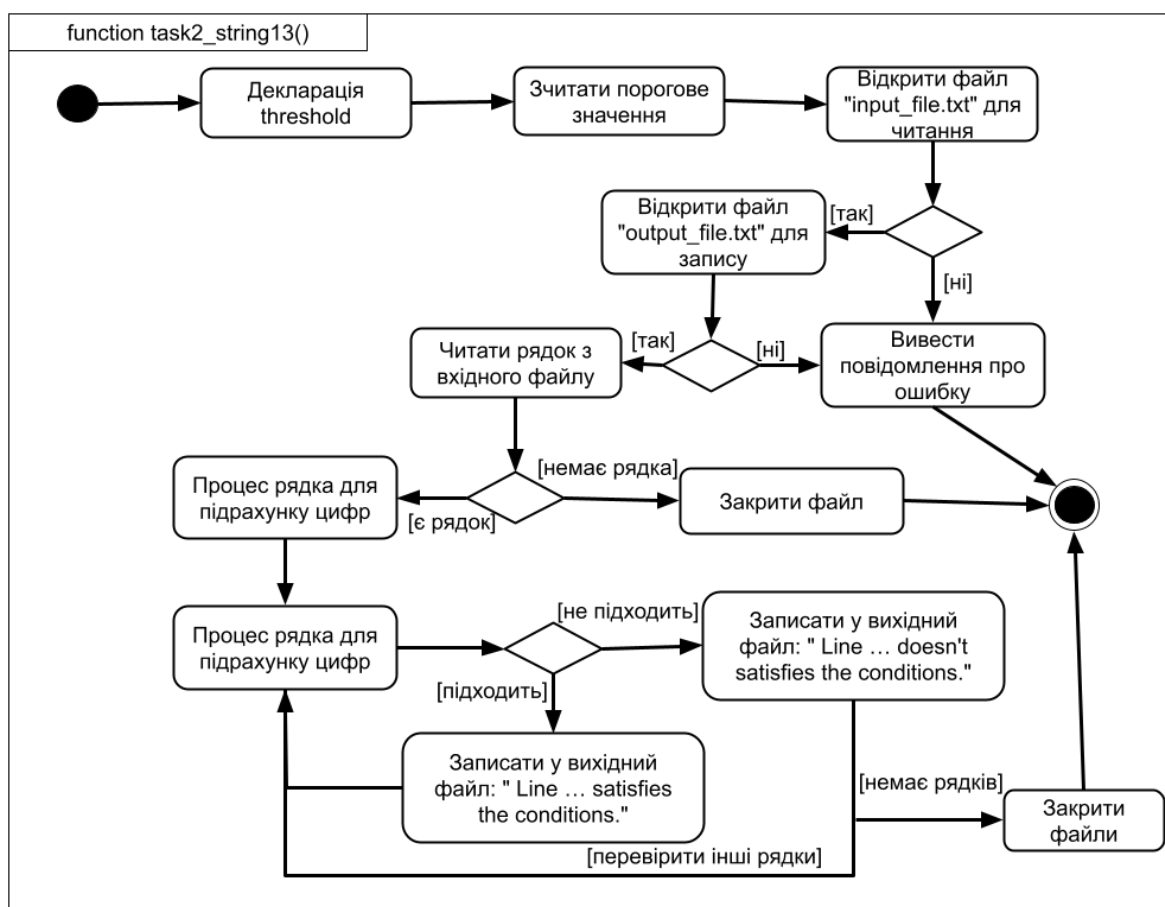


Рисунок 2 — Алгоритм вирішення String13

Лістинг коду вирішення задачі String13 наведено в дод. А (стор. 7).

Екран роботи програми показаний на рис. Б.2-3.

ВИСНОВКИ

Вивчено теоретичний матеріал з основ роботи з низькорівневими рядками на C++ і документацію до класу `string`, а також алгоритми пошуку в рядку, а також реалізовано обробку рядків на C++. Відпрацьовано в коді виведення та виведення даних у файл. Виникли труднощі з функцією `compare`.

ДОДАТОК А

Лістинг коду програми

```

#include <iostream>
#include <fstream>
#include <string>
using namespace std;

void task1_45();
int compare(const char* source, int pos, int len, const char* s);

void task2_string13();
int processString(string input_string);
bool satisfiesConditions(string input_string, int threshold);

int main() { // menu
    cout << " " << endl;
    int task_num; // diclaration
    do {
        cout << " Task number (0 - exit): ";
        cin >> task_num;
        if (!cin) {
            cout << " Ups!" << endl; // error notification
            continue;
        }
        switch (task_num) {
            case 1 : task1_45(); break; // task 1
            case 2 : task2_string13(); break; // task 2
            case 0 : cout << " The program is over!" << endl; break;
            default : cout << " Wrong task number!" << endl;
                    // output for incorrect numbers
        }
    } while (task_num != 0); // end the program
    return 0;
}

// task 1
void task1_45() {
    cout << " ### Task 1 - 45 ### " << endl;
    string str1 ("wonderful day"); // creating string str1
    const char * str2 = "wonderful day"; // creating a pointer to the string
    str2
    int comp_res;
    // comparing part of the string str1 with a given string
    comp_res = str1.compare(10, 4, "day");
    if (comp_res == 0)
        cout << " We need " << str1 << " every day!" << endl;
        // display a message about string equality
    else
        cout << " Not equal: " << comp_res << endl;
}

```

```

        // display a message about string inequality
        // comparing part of the string str2 with a given string
        comp_res = compare(str2, 10, 4, "day");
        if (comp_res == 0)
            cout << " We need " << str2 << " every day!" << endl;
            // display a message about string equality
        else
            cout << " Not equal: " << comp_res << endl;
            // display a message about string inequality

        return;
    }
    // function for comparing parts of strings
    int compare(const char* source, int pos, int len, const char* s) {
        for (int i = 0; i < len; i++) { // loop through characters in part of a
            string
                if (source[pos + i] != s[i])
                    return source[pos + i] - s[i];
                    // return the difference between character codes
        }
        return 0;
    }
    // task 2
    void task2_string13() {
        cout << " ### Task 2 String13 ### " << endl;

        int threshold; // declaration of a variable for the threshold value
        cout << " Enter the threshold value: ";
        cin >> threshold;

        ifstream input_file("input_file.txt"); // opening a file for reading
        ofstream output_file("output_file.txt"); // opening a file for writing

        if (!input_file) {
            cout << " Could not open input file." << endl;
            // display an error message
            return;
        }
        if (!output_file) {
            cout << " Could not open output file." << endl;
            // display an error message
            return;
        }
        // declaration of a variable for a file line
        string line;
        while (getline(input_file, line)) {
            if (satisfiesConditions(line, threshold)) {
                output_file << " Line '" << line << "' satisfies the conditions." <<
endl;
                // recording a satisfaction message
            }
        }
    }
}

```

```

        } else {
            output_file << " Line '" << line << "' doesn't satisfy the
conditions." << endl;
            // recording a message about non-fulfillment of conditions
        }
    }

    input_file.close(); // close the file for reading
    output_file.close(); // close the file for writing
    cout << " Answer is in the 'output_file.txt' " << endl;
    // displaying a message about the location of the result
    return;
}

// function to process a string and count digits
int processString(string input_string) {
    int digit_count = 0; // declaration of a variable to count digits
    for (char c : input_string) {
        if (isdigit(c)) {
            digit_count++; // increasing digit counter
        }
    }
    return digit_count; // return number of digits
}

// function to check a string to see if it meets the conditions
bool satisfiesConditions(string input_string, int threshold) {
    int digit_count = processString(input_string); // counting the number of
digits in a string
    return (digit_count >= threshold);
    // returning the result of checking for compliance with the threshold value
}

```


ДОДАТОК Б

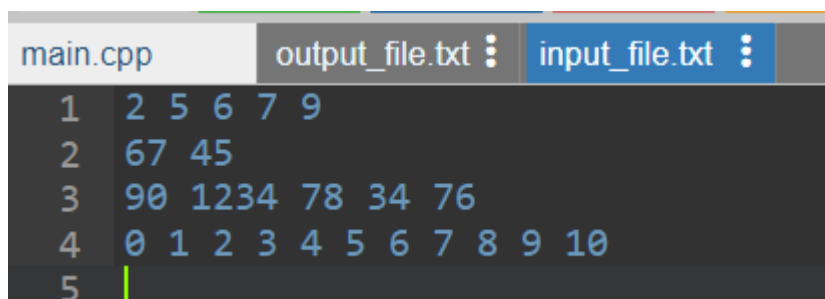
Скрін-шоти вікна виконання програми

```

Task number (0 - exit): 1
### Task 1 - 45 ###
We need wonderful day every day!
We need wonderful day every day!
Task number (0 - exit): 2
### Task 2 String13 ###
Enter the threshold value: 3
Answer is in the 'output_file.txt'
Task number (0 - exit): 0
The program is over!

```

Рисунок Б.1 – Екран виконання програми для вирішення завдання
Завдання 1-45

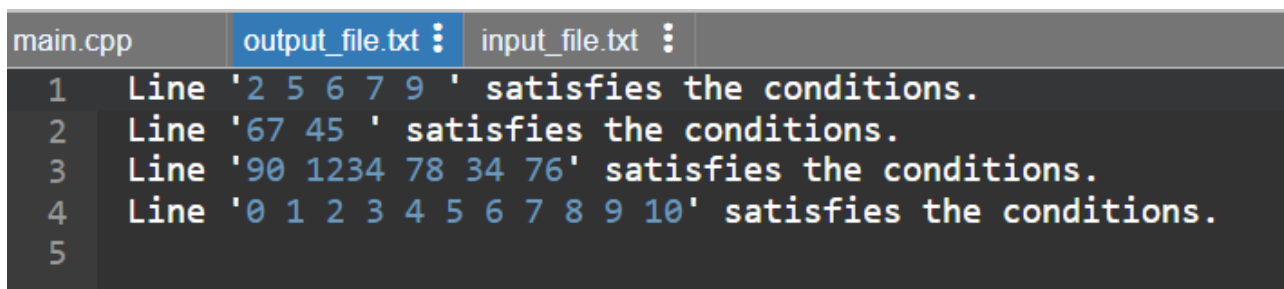


```

main.cpp  output_file.txt  input_file.txt
1  2 5 6 7 9
2  67 45
3  90 1234 78 34 76
4  0 1 2 3 4 5 6 7 8 9 10
5

```

Рисунок Б.2 – Екран виконання програми для вирішення завдання
String13



```

main.cpp  output_file.txt  input_file.txt
1  Line '2 5 6 7 9 ' satisfies the conditions.
2  Line '67 45 ' satisfies the conditions.
3  Line '90 1234 78 34 76' satisfies the conditions.
4  Line '0 1 2 3 4 5 6 7 8 9 10' satisfies the conditions.
5

```

Рисунок Б.3 – Екран виконання програми для вирішення завдання
String13