

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний інститут
імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи №1 з дисципліни
« Основи програмування 2. Модульне програмування»

«Текстові файли»
Варіант 3

Виконав студент ІП-15, Борисик Владислав Тарасович
(шифр, прізвище, ім'я, по батькові)

Перевірила Вечерковська Анастасія Сергіївна
(прізвище, ім'я, по батькові)

Київ 2022

Лабораторна робота №1

Текстові файли

Варіант 3

Задача

3. Створити текстовий файл. Слова у файлі відділені пробілами, комами, крапками. У кожному реченні визначити саме довге слово. Створити новий текстовий файл, у якому кожне речення розміщується у окремому рядку і починається із числа, що дорівнює довжині самого довгого слова в ньому, за яким йде саме це слово. Вивести вміст вихідного і створеного файлів.

Код

C++

main.cpp:

```
#include "functions.h"
using namespace std;

int main() {
    // читаємо текст з консолі, поки не буде натиснуто комбінацію Ctrl+D
    string text_to_write = capture_text();
    // записуємо цей текст у файл input.txt
    write_empty_file("input.txt", text_to_write);

    // читаємо текст з файлу input.txt у змінну text
    string text = read_input_file("input.txt");
    printf("\nText on input:\n%s\n\n", text.c_str());

    // вектор з реченнями тексту
    vector<string> sentences = split_sentences(text);
    // виводимо елементи вектора в консоль
    print_vector(sentences);

    // вектор з найдовшими словами кожного речення
    vector<string> longest_words = get_longest_words(sentences);
    // виводимо найдовші слова в консоль
    print_longest_words(longest_words);

    // записуємо текст у файл output.txt
    write_file("output.txt", longest_words, sentences);
    // читаємо записаний текст з файлу output.txt
    read_output_file("output.txt");
}
```

functions.h:

```
#ifndef INC_2LABWORK_1_FUNCTIONS_H
#define INC_2LABWORK_1_FUNCTIONS_H
#endif

#include <iostream>
#include <fstream>
#include <vector>

using namespace std;

/** Отримує текст з клавітури, поки не натиснута комбінація клавіш Ctrl+Enter
 * @return: текст з клавітури
 */
string capture_text();

/** Виводить елементи вектора в консоль
 * @param vec: вектор
 */
void print_vector(const vector<string>& vec);

/** Читає текст з вхідного файлу
 * @param file_name: назва файлу
 * @return: текст з файлу
 */
string read_input_file(const string& file_name);

/** Ділить речення у тексті у вектор
 * @param text: текст
 * @return: вектор з реченнями
 */
vector<string> split_sentences(const string& text);

/** Виводить в консоль найдовші слова і їх довжину
 * @param longest_words: вектор
 */
void print_longest_words(const vector<string>& longest_words);

/** Отримує найдовші слова із вектора речень
 * @param sentences: вектор з реченнями
 * @return: вектор з найдовшими словами
 */
vector<string> get_longest_words(const vector<string>& sentences);

/** Записує текст у файл
 * @param file_name: назва файлу
 * @param longest_words: вектор з найдовшими словами речень
 * @param sentences: вектор з реченнями
 */
void write_file(const string& file_name, const vector<string>& longest_words,
vector<string>& sentences);

/** Записує текст у файл input
 * @param file_name: назва файлу
```

```
* @param text: текст
*/
void write_empty_file(const string& file_name, const string& text);

/** Читає текст з вихідного файлу
 * @param file_name: назва файлу
 */
void read_output_file(const string& file_name);
```

functions.cpp:

```
#include "functions.h"

string capture_text(){
    cout << "Enter your text\nPress Enter to go to the next line\nPress Ctrl +
D - to end writing\n";

    string text;
    string line;
    // ascii код комбінації клавіш Ctrl + D
    int ascii_code = 94;
    // поки не натиснута комбінація клавіш Ctrl + D
    while(int(line[0]) != ascii_code){
        getline(cin, line);
        text += line + '\n';
    }
    // відкидаємо 4 зайвих символи (2 зайвих \n і ^D)
    text = text.substr(0, text.size()-4);

    return text;
}

void write_empty_file(const string& file_name, const string& text){
    // потік write_file (для читання і запису)
    ofstream write_file(file_name);
    // записуємо текст у файл
    write_file << text;
    // закриваємо потік(файл)
    write_file.close();
}

void print_vector(const vector<string>& vec){
    cout << "Sentences vector:\n";

    for(auto & i : vec){
        cout << i << "\n";
    }
    cout << "\n";
}

string read_input_file(const string& file_name){
    // потік read_file (тільки для читання)
    ifstream read_file(file_name);
    string text;
    string line;

    // поміщаємо текст з потоку у змінну line
    while (getline(read_file, line)){
        text += line + '\n';
    }
}
```

```

// закриваємо потік(файл)
read_file.close();

// відкидуємо зайвий символ
text = text.substr(0, text.size()-1);

return text;
}

vector<string> split_sentences(const string& text){
    // вектор з реченнями
    vector<string> sentences;
    // позиція минулої крапки
    int previous_position = 0;
    // лічильник (рахує символи тексту)
    int counter = 0;

    while(counter <= text.size()){
        // якщо символ дорівнює крапці або (counter дорівнює останньому символу
        // тексту і минулий символ не дорівнює крапці)
        // (counter == text.size() && text[counter-1] != '.') потрібно для
        // того, що б цикл коректно виділив останнє речення, якщо воно записане без
        // крапки
        if(text[counter] == '.' || text[counter] == '\n' || (counter ==
        text.size() && text[counter-1] != '.')){
            // речення (підрядок, який починається з позиції минулої крапки
            // довжиною counter-previous_position)
            string sentence = text.substr(previous_position, counter-
            previous_position);
            // додаємо це речення у вектор
            sentences.push_back(sentence);
            // оновлюємо позицію крапки
            previous_position = counter+1;
        }
        counter++;
    }

    return sentences;
}

vector<string> split_words(const vector<string>& sentences, int i){
    // вектор зі словами
    vector<string> words;
    // позиція минулої крапки
    int previous_position = 0;
    // лічильник (рахує символи тексту)
    int counter = 0;

    while(counter <= sentences[i].size()){
        // якщо символ дорівнює розділювачу або останньому символу слова
        if(sentences[i][counter] == ',' || sentences[i][counter] == ' ' ||

```

```

counter == sentences[i].size()){
    // слово (підрядок, який починається з позиції минулого розділювача
довжиною counter-previous_position)
    string word = sentences[i].substr(previous_position, counter-
previous_position);
    // додаємо це слово у вектор
words.push_back(word);
    // оновлюємо позицію
previous_position = counter+1;
}
counter++;
}

return words;
}

void print_longest_words(const vector<string>& longest_words){
    for(int i =0; i < longest_words.size(); i++){
        printf("Longest word in sentence #%d is: \"%s\" and its length: %d\n",
i+1, longest_words[i].c_str(), longest_words[i].length());
    }
    cout << "\n";
}

vector<string> get_longest_words(const vector<string>& sentences){
    // вектор з найдовшими словами
vector<string> longest_words;

    for(int i=0; i < sentences.size(); i++){
        // ділимо речення на слова
vector<string> words = split_words(sentences, i);
        // найдовше слово (перше слово у реченні)
string longest_word = words[0];

        for(int j=1; j<words.size();j++){
            // якщо довжина слова більша за довжину longest_word
            if(words[j].size() > longest_word.size()){
                // оновлюємо найдовше слово
                longest_word = words[j];
            }
        }
        // додаємо найдовше слово у вектор
        longest_words.push_back(longest_word);
    }

    return longest_words;
}

void write_file(const string& file_name, const vector<string>& longest_words,
vector<string>& sentences){

```

```
// потік write_file (для читання і запису)
ofstream write_file(file_name);
// записуємо текст у файл
for(int i=0; i < sentences.size(); i++){
    write_file << to_string(longest_words[i].size()) + " " +
longest_words[i] + " | " + sentences[i] + "\n";
}
// закриваємо потік(файл)
write_file.close();
}

void read_output_file(const string& file_name){
    // потік read_output_file (тільки для читання)
    ifstream read_output_file(file_name);
    // виводимо в консоль текст з файлу
    cout << read_output_file.rdbuf() << "\n";
    // закриваємо потік(файл)
    read_output_file.close();
}
```


Python

main.py:

```
from functions import *

# читаємо текст з консолі, поки не буде натиснуто клавішу Ctrl + D
text_to_write = capture_text()
# записуємо цей текст у файл input.txt
write_empty_file("input.txt", text_to_write)

# читаємо текст з файлу input.txt у змінну text
text = read_file("input.txt")
print(f"Text on input:\n{text}\n")

# список з реченнями тексту
sentences = split_sentences(text)
print("Sentences list:")
print(*sentences, sep='\n')

# вектор з найдовшими словами кожного речення
longest_words = get_longest_words(sentences)
print_longest_words(longest_words)

# записуємо текст у файл output.txt
write_to_file("output.txt", longest_words, sentences)
output_text = read_file("output.txt")
print(output_text)
```

functions.py:

```
def capture_text():
    """
    Читає текст з консолі, поки не буде натиснуто клавішу Esc
    :return: текст з консолі
    """
    print('Enter your text\nPress Enter to go to the next line\nPress Ctrl + D - to
end writing')

    import sys
    text = sys.stdin.read()

    # відкидаємо зайвий символ \n
    text = text[:len(text) - 1]

    print(f"Whole text:\n{text}\n")

    return text


def write_empty_file(file_name: str, text: str):
    """
    Записуємо текст у файл
    :param file_name: назва файлу
    :param text: текст
    """
    write_file = open(file_name, 'wt')
    write_file.write(text)
    write_file.close()


def get_longest_words(sentences: list[str]):
    """
    Отримує найдовші слова із списку речень
    :param sentences: список з реченнями
    :return: список з найдовшими словами
    """
    # список з найдовшими словами
    longest_words = []

    for i in range(len(sentences)):
        # заміняємо у реченні коми на пробіли (потрібно для того, щоб коректно
        # відділити слова)
        new_sentence = sentences[i].replace(',', ' ')
        # список із словами
        words = new_sentence.split(' ')
        # найдовше слово (перше слово у реченні)
        longest_word = words[0]

        for j in range(1, len(words)):
            # якщо довжина слова більша за довжину longest_word
            if len(words[j]) > len(longest_word):
                longest_word = words[j]
```

```

        # додаємо найдовше слово у вектор
        longest_words.append(longest_word)

    return longest_words

def print_longest_words(longest_words: list[str]):
    """
    Виводить в консоль найдовші слова і їх довжину
    :param longest_words: список із словами
    """
    print('\n')
    for i in range(len(longest_words)):
        print(f'Longest word in sentence #{i+1} is: "{longest_words[i]}" and its
length: {len(longest_words[i])}')
    print('\n')

def split_sentences(text: str):
    """
    Ділить речення у тексті
    :param text: текст
    :return: список з реченнями
    """
    text = text.replace('\n', '.')
    sentences = text.split('.')

    for i in range(len(sentences)):
        # перевіряємо, чи є у списку "пусті речення" (потрібно для того, щоб у список
не заносився пустий елемент)
        if sentences[i] == '':
            sentences.pop(i)

    return sentences

def write_to_file(file_name: str, longest_words: list[str], sentences: list[str]):
    """
    Записує текст у файл
    :param file_name: назва файлу
    :param longest_words: список з найдовшими словами речень
    :param sentences: список з реченнями
    :return:
    """
    write_file = open(file_name, 'wt')

    for i in range(len(longest_words)):
        write_file.write(f"{len(longest_words[i])} {longest_words[i]} |
{sentences[i]}\n")

    write_file.close()

def read_file(file_name: str):

```

```
"""
Читає текст з файлу
:param file_name: назва файлу
:return: текст з файлу
"""

read_input_file = open(file_name, 'rt')
text = read_input_file.read()
read_input_file.close()

return text
```

Результат виконання програми

C++:

```
Enter your text
Press Enter to go to the next line
Press Ctrl + D - to end writing
Hello words
One sentence.Some spaces h.One,comma
32131 s231,3123132
^D

Text on input:
Hello words
One sentence.Some spaces h.One,comma
32131 s231,3123132

Sentences vector:
Hello words
One sentence
Some spaces h
One,comma
32131 s231,3123132

Longest word in sentence #1 is: "Hello" and its length: 5
Longest word in sentence #2 is: "sentence" and its length: 8
Longest word in sentence #3 is: "spaces" and its length: 6
Longest word in sentence #4 is: "comma" and its length: 5
Longest word in sentence #5 is: "3123132" and its length: 7
```

```
5 Hello | Hello words
8 sentence | One sentence
6 spaces | Some spaces h
5 comma | One,comma
7 3123132 | 32131 s231,3123132
```

input.txt – Блокнот

Файл Правка Формат Вид Справка

```
Hello words
One sentence.Some spaces h.One,comma
32131 s231,3123132
```

output.txt – Блокнот

Файл Правка Формат Вид Справка

```
5 Hello | Hello words
8 sentence | One sentence
6 spaces | Some spaces h
5 comma | One,comma
7 3123132 | 32131 s231,3123132
```

Python:

```
Enter your text
Press Enter to go to the next line
Press Ctrl + D - to end writing
Hello words
Another sentence.One space.And,some,commas
32131 s231,3123132
^D
Whole text:
Hello words
Another sentence.One space.And,some,commas
32131 s231,3123132

Text on input:
Hello words
Another sentence.One space.And,some,commas
32131 s231,3123132

Sentences list:
Hello words
Another sentence
One space
And,some,commas
32131 s231,3123132
```

```
Longest word in sentence #1 is: "Hello" and its length: 5
Longest word in sentence #2 is: "sentence" and its length: 8
Longest word in sentence #3 is: "space" and its length: 5
Longest word in sentence #4 is: "commas" and its length: 6
Longest word in sentence #5 is: "3123132" and its length: 7
```

```
5 Hello | Hello words
8 sentence | Another sentence
5 space | One space
6 commas | And,some,commas
7 3123132 | 32131 s231,3123132
```



input.txt – Блокнот

Файл Правка Формат Вид Справка

```
Hello words
Another sentence.One space.
32131 s231,3123132
```



output.txt – Блокнот

Файл Правка Формат Вид Справка

```
5 Hello | Hello words
8 sentence | Another sentence
5 space | One space
6 commas | And,some,commas
7 3123132 | 32131 s231,3123132
```