

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 18

Виконав студент

Лазьов Кирило Владиславович
(шифр, прізвище, ім'я, по батькові)

Перевірила

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

Лабораторна робота №1
Текстові файли
Варіант 18
Задача

18. Створити текстовий файл. Сформуванати новий текстовий файл, що складається з рядків вихідного файлу, у яких вилучені всі двосимвольні слова. Визначити і дописати в кінець нового файлу кількість вилучених слів. Вивести вміст вихідного і створеного файлів.

Код

C++:

main.cpp:

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

int main() {
    string text_to_write = get_text();

    write_empty_file("input.txt", text_to_write);

    string text = read_input("input.txt");

    printf("\nEntered Text:\n%s\n\n", text.c_str());

    vector<string> sentences = split_sentences(text);

    string NewText = ChangeText(sentences);

    write_file("output.txt", NewText, sentences);

    read_output("output.txt");
}
```

Header.h:

```
#include <iostream>
#include <fstream>
#include<string>
#include <vector>

using namespace std;

string get_text();

string read_input(const string& file_name);

vector<string> split_sentences(const string& text);

string ChangeText(const vector<string>& sentences);

void write_file(const string& file_name, const string& NewText, vector<string>& sentences);

void write_empty_file(const string& file_name, const string& text);

void read_output(const string& file_name);
```

Functions.cpp:

```
#include "Header.h"

string get_text() {
    cout << "Enter your text. Enter to go to the next line. Ctrl + D - end writing" << endl;
    string text;
    string line;
    int ascii_code = 4;
    while (int(line[0]) != ascii_code) {
        getline(cin, line);
        text += line + '\n';
    }
    text = text.substr(0, text.size() - 3);
    return text;
}

void write_empty_file(const string& file_name, const string& text) {
    ofstream write_file(file_name);
    write_file << text;
    write_file.close();
}

string read_input(const string& file_name) {
    ifstream read_file(file_name);
    string text;
    string line;
    while (getline(read_file, line)) {
        text += line + '\n';
    }
    read_file.close();
    text = text.substr(0, text.size() - 1);
    return text;
}
```

```
vector<string> split_sentences(const string& text) {  
    vector<string> sentences;  
    int prev_pos = 0;  
    int counter = 0;  
  
    while (counter <= text.size()) {  
        if (text[counter] == '\\n' || counter == text.size()) {  
            string sentence = text.substr(prev_pos, counter - prev_pos);  
            sentences.push_back(sentence);  
            prev_pos = counter + 1;  
        }  
        counter++;  
    }  
  
    return sentences;  
}  
  
vector<string> split_words(const vector<string>& sentences, int i) {  
    vector<string> words;  
    int prev_pos = 0;  
    int counter = 0;  
  
    while (counter <= sentences[i].size()) {  
        if (sentences[i][counter] == ' ' || counter == sentences[i].size()) {  
            string word = sentences[i].substr(prev_pos, counter - prev_pos);  
            words.push_back(word);  
            prev_pos = counter + 1;  
        }  
        counter++;  
    }  
  
    return words;  
}
```

```

string ChangeText(const vector<string>& sentences) {
    string NewText;
    int counter = 0;
    for (int i = 0; i < sentences.size(); i++) {
        vector<string> words = split_words(sentences, i);
        for (int j = 0; j < words.size(); j++) {
            if (words[j].size() != 2) {
                NewText += words[j] + " ";
            }
            else {
                counter++;
            }
        }
        NewText += "\n";
    }

    NewText += to_string(counter) + " words was deleted";

    return NewText;
}

void write_file(const string& file_name, const string& NewText, vector<string>& sentences) {
    ofstream write_file(file_name);
    write_file << NewText + "\n";
    write_file.close();
}

void read_output(const string& file_name) {
    ifstream read_output_file(file_name);
    cout << read_output_file.rdbuf() << "\n";
    read_output_file.close();
}

```

Python:

main.py:

```
import func

text_to_write = func.get_text()

func.write_empty_file("input.txt", text_to_write)

text = func.read_file("input.txt")

print("Entered text:", "\n", text)

NewText = func.ChangeText(text)

func.write_to_file("output.txt", NewText)

output_text = func.read_file("output.txt")

print("\n", "Changed text:", "\n", output_text)
```

func.py

```
def get_text():
    print('Enter your text. Enter to go to the next line. Ctrl + D - end writing')

    import sys
    text = sys.stdin.read()

    text = text[:len(text) - 1]

    return text
```

```
def write_empty_file(file_name: str, text: str):

    write_file = open(file_name, 'wt')
    write_file.write(text)
    write_file.close()
```

```
def ChangeText(text: str):
    NewText = ""
    count = 0
    sentences = text.split("\n")
    for i in range(len(sentences)):
        sentence = sentences[i]
        words = sentence.split(" ")
        for j in range(len(words)):
            if len(words[j]) != 2:
                NewText += words[j] + " "
            else:
                count += 1
        NewText += "\n"
    NewText += str(count) + " words was removed"
    return NewText
```

```
def write_to_file(file_name: str, NewText):
    write_file = open(file_name, 'wt')

    write_file.write(NewText)

    write_file.close()
```

```
def read_file(file_name: str):
    read_input_file = open(file_name, 'rt')
    text = read_input_file.read()
    read_input_file.close()

    return text
```


Результат виконання

C++:

```
Enter your text. Enter to go to the next line. Ctrl + D - end writing
so siad iwaji 11 sajd pjwaj lo
diwaoid spd 9wa odsjaj lo dwa jdsah 22
^D
```


```
Entered Text:
so siad iwaji 11 sajd pjwaj lo
diwaoid spd 9wa odsjaj lo dwa jdsah 22
```

```
siad iwaji sajd pjwaj
diwaoid spd 9wa odsjaj dwa jdsah
5 words was deleted
```

 input.txt - Notepad

File Edit Format View Help

```
so siad iwaji 11 sajd pjwaj lo
diwaoid spd 9wa odsjaj lo dwa jdsah 22
```

 output.txt - Notepad

File Edit Format View Help


```
siad iwaji sajd pjwaj
diwaoid spd 9wa odsjaj dwa jdsah
5 words was deleted
```

Python:

```
Enter your text. Enter to go to the next line. Ctrl + D - end writing
so sahd oiwhaoi lo 22
pod iohawio dshaiohd ioas 11 lo po
^D
```


```
Entered text:
so sahd oiwhaoi lo 22
pod iohawio dshaiohd ioas 11 lo po
```

```
Changed text:
sahd oiwhaoi
pod iohawio dshaiohd ioas
6 words was removed
```


 input.txt - Notepad

File Edit Format View Help

```
so sahd oiwhaoi lo 22  
pod iohawio dshaiodh ioas 11 lo po
```

 output.txt - Notepad

File Edit Format View Help

```
sahd oiwhaoi  
pod iohawio dshaiodh ioas  
6 words was removed
```