

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Катедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 1 з дисципліни

«Основи програмування 2.

Модульне програмування»

«Файли даних»

Варіант 18

Виконав студент ІП-11 Лесів Владислав Ігорович

Перевірів Вітковська Ірина Іванівна

Київ 2022

Лабораторна робота 1

Файли даних

Мета – вивчити особливості створення і обробки текстових файлів даних.

Варіант №18.

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

Постановка задачі. Результатом розв’язку є текстовий файл, у якому видалено двосимвольні слова і визначено їх кількість так, як це вказано в умові задачі відповідно до заданого файлу. Для визначення результату повинен бути заданий текстовий файл зі якимось текстом усередині. Інших початкових даних для розв’язку не потрібно.

Математичне формулювання задачі зводиться до знаходження двосимвольних слів, якщо попередній символ від першого в слові є пробілом, а наступний після другого символу також пробіл; додавання до загальної кількості знайдених слів.

Виконання мовою C++.

Код програми:

files.h

```
#pragma once
#include <iostream>
#include <fstream>
#include <string>
#include <conio.h>

using namespace std;

void input(int); //Введення початкового тексту
void writing(string, string); //Запис у вихідний файл
string processLine(string, int*); //Обробка рядків
void output(string); //Вивід на консоль
```

Lab1_cpp.cpp

```
#include "files.h"

int main()
{
    int act;
    cout << "Clear (0) or append (1)? : ";    //Додавання до файлу чи очищення
    cin >> act;
    input(act);
    writing("input.txt", "output.txt");
    cout << "Entered file:\n";
    output("input.txt");
    cout << "\nCreated file:\n";
    output("output.txt");
    return 0;
}
```

files.cpp

```
#include "files.h"

string processLine(string line, int* count) {
    line += " ";
    string lineOut = "",
           lineTemp = "";
    int i = 0, num = 0;
    while (i < line.length()) {
        if (line[i] != ' ') {                //Формуємо слова
            num++;
            lineTemp += line[i];
        }
        else {
            if (num == 2) {                  //Враховуємо слова з двох символів
                (*count)++;
                cout << lineTemp << " ";
            }
            else {
                lineOut += lineTemp;
                lineOut += " ";
            }
            num = 0;
            lineTemp = "";
        }
        i++;
    }
    return lineOut;
}

void output(string name) {
    ifstream inFile(name);
    string line;
    while (!inFile.eof()) {
        getline(inFile, line);
        cout << line << endl;
    }
    inFile.close();
}

void input(int act) {
    fstream inFile;
    if (act == 0) {                          //Якщо задано очищення вхідного файлу
        inFile.open("input.txt", ios::out);
        inFile.close();
    }
}
```

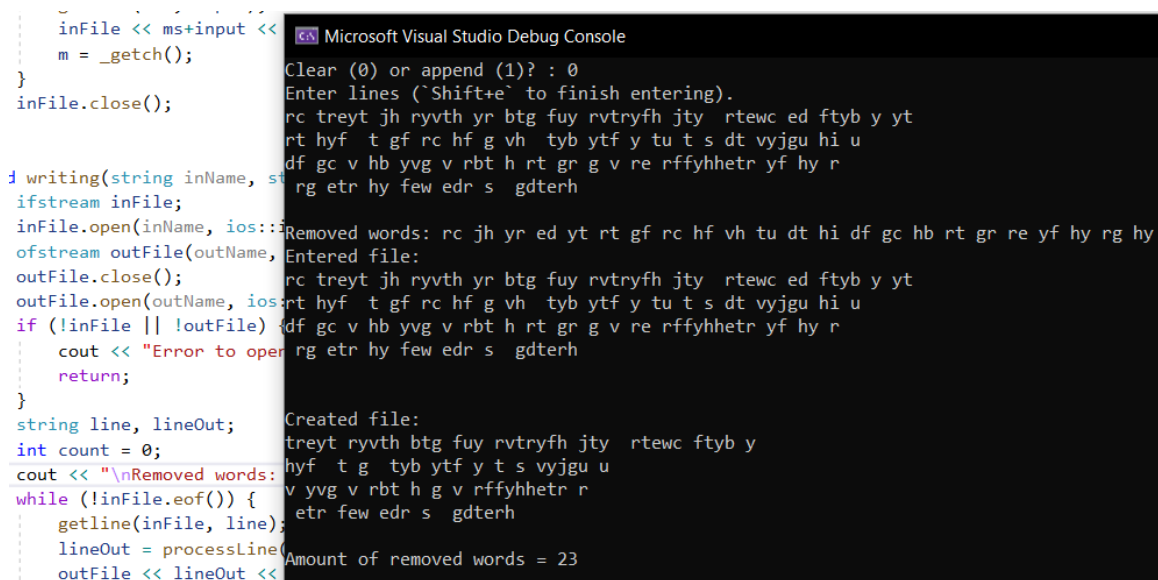
```

inFile.open("input.txt", ios::app);
if (!inFile) {
    cout << "Can't open input file.";
    return;
}
string input, ms;
cout << "Enter lines (`Shift+e` to finish entering)." << endl;
char m = _getch();
cin.ignore();
while (m != 'E') {
    cout << m;
    ms = m;
    getline(cin, input);
    inFile << ms+input << endl;
    m = _getch();
}
inFile.close();
}

void writing(string inName, string outName) {
    ifstream inFile;
    inFile.open(inName, ios::in);
    ofstream outFile(outName, ios::out);
    outFile.close();
    outFile.open(outName, ios::app);
    if (!inFile || !outFile) {
        cout << "Error to open file(s)";
        return;
    }
    string line, lineOut;
    int count = 0;
    cout << "Removed words: ";
    while (!inFile.eof()) {
        getline(inFile, line);
        lineOut = processLine(line, &count);
        outFile << lineOut << endl;
    }
    cout << endl;
    outFile << "Amount of removed words = " << count;
    inFile.close();
    outFile.close();
}

```

Випробування алгоритму.



The screenshot shows the execution of the C++ program. The input file 'input.txt' contains the following text:

```

rc treyt jh ryvth yr btg fuy rvtryfh jty rtewc ed ftyb y yt
rt hyf t gf rc hf g vh tyb ytf y tu t s dt vyjgu hi u
df gc v hb yvg v rbt h rt gr g v re rffyhhetr yf hy r
rg etr hy few edr s gdterh

```

The program processes this text and creates an output file 'output.txt' containing the text with removed words:

```

treyt ryvth btg fuy rvtryfh jty rtewc ftyb y
hyf t g tyb ytf y t s vyjgu u
v yvg v rbt h g v rffyhhetr r
etr few edr s gdterh

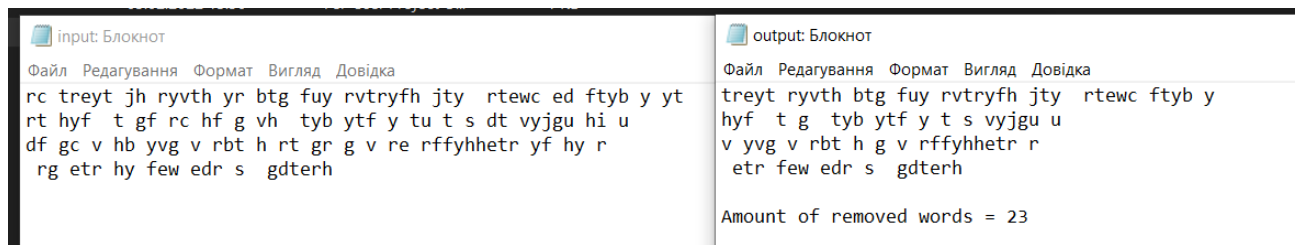
```

The console output shows the following messages:

```

Clear (0) or append (1)? : 0
Enter lines (`Shift+e` to finish entering).
Removed words: rc jh yr ed yt rt gf rc hf vh tu dt hi df gc hb rt gr re yf hy rg hy
Entered file:
rc treyt jh ryvth yr btg fuy rvtryfh jty rtewc ed ftyb y yt
rt hyf t gf rc hf g vh tyb ytf y tu t s dt vyjgu hi u
df gc v hb yvg v rbt h rt gr g v re rffyhhetr yf hy r
rg etr hy few edr s gdterh
Created file:
treyt ryvth btg fuy rvtryfh jty rtewc ftyb y
hyf t g tyb ytf y t s vyjgu u
v yvg v rbt h g v rffyhhetr r
etr few edr s gdterh
Amount of removed words = 23

```



Виконання мовою Python.

Код програми:

Lab1_py.py

```
from fileModule import *

act=int(input("Clear (0) or append (1)? : "))
enter(act)
writing("input.txt","output.txt")
print("Entered file:")
output("input.txt")
print("\nCreated file:")
output("output.txt")
```

#Додавання до файлу чи очищення

fileModule.py

```
import msvcrt

def processLine(lines):
    coun=0
    for i in range(len(lines)):
        lines[i]=lines[i].split()
        lineTemp=""
        for k in lines[i]:
            if len(k)==2:
                count+=1
                print(k,end=" ")
            else:
                lineTemp+=k
                lineTemp+=" "
        lines[i]=lineTemp
    outText="\n".join(lines)
    return outText, coun

def output(name):
    with open(name) as file:
        for i in file.readlines():
            print(i, end="")

def enter(act):
    if act==0:
        with open("input.txt","w") as inFile:
            pass
        with open("input.txt","a") as inFile:
            print("Enter lines (`shift+e` to finish entering).")
            m = msvcrt.getch().decode('ASCII')
            while m!='E':
                #Якщо задано очищення вхідного файлу
                #Перевіряємо, чи введений shift+e
```

```

        print(m,end="")
        inp=input()
        inFile.write(m+inp+"\n")
        m = msvcrt.getch().decode('ASCII')

def writing(inName,outName):
    with open(inName,"r") as inFile:
        with open(outName,"w") as outFile:
            outFile.write("")
        with open(outName,"a") as outFile:
            print("\nRemoved words: ", end="")
            line=inFile.readlines()
            lineOut, coun=processLine(line)    #Оброблюємо текст відповідно до умови
            print()
            outFile.write(lineOut)
            outFile.write("\nAmount of removed words = "+str(coun)+"\n")

```

Випробування алгоритму.

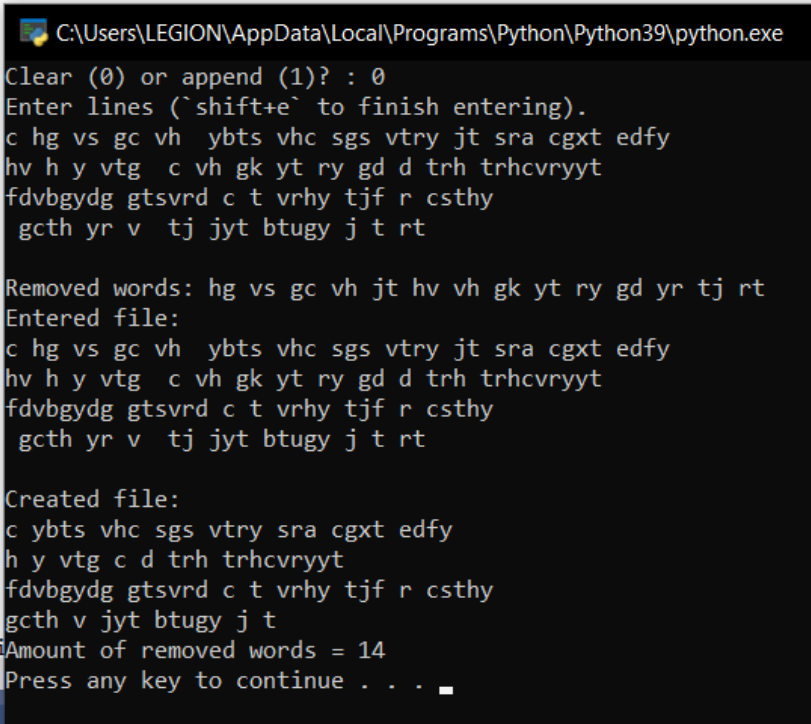
```

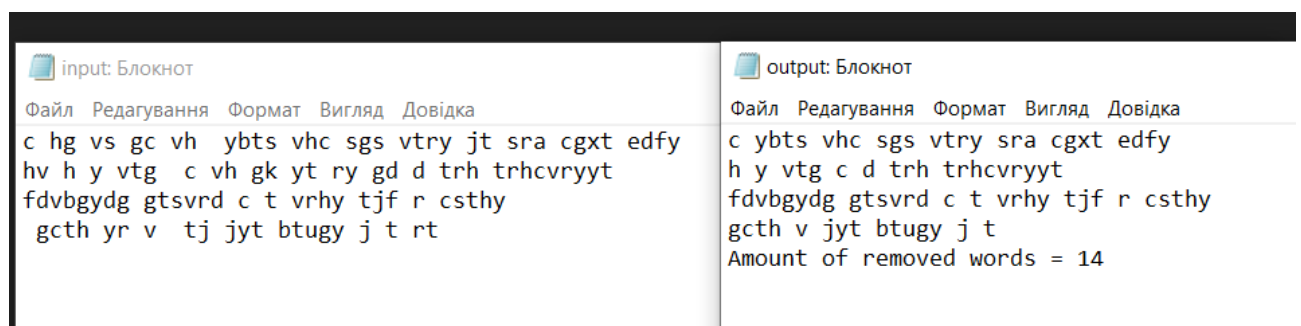
lines[i]=lines[i].split()
lineTemp=""
for k in lines[i]:
    if len(k)==2:
        coun+=1
        print(k,end=" ")
    else:
        lineTemp+=k
        lineTemp+=" "
lines[i]=lineTemp
outText="\n".join(lines)
return outText, coun

output(name):
with open(name) as file:
    for i in file.readlines():
        print(i, end="")

enter(act):
if act==0:
    with open("input.txt","w") as i

```





Висновок. Отже, у цій роботі я вивчив особливості створення і обробки текстових файлів даних. У результаті лабораторної роботи було розроблено програми, які виконують задачу відповідно до постановки. Використовуючи роботу з файлами для читання та запису необхідної інформації, функції для

обробки тексту, прочитаного з файлу, за умовою задачі та виведення тексту з файлів на консоль, отримуємо коректний результат.