

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 7

Виконав студент ІП-15, Гуменюк Олександр Володимирович

(шифр, прізвище, ім'я, по батькові)

Перевірів Вєчерковська Анастасія Сергіївна

(прізвище, ім'я, по батькові)

Київ 2022

Лабораторна робота 1

Текстові файли

Індивідуальне завдання

Варіант 7

7. Створити текстовий файл. Сформувати новий текстовий файл, що складається зі слів вхідного файлу, які зустрічаються у ньому більше N раз. Розмістити ці слова в новому файлі в порядку зростання їхньої довжини. Вивести вміст вихідного і створеного файлів.

Код C++

[OP_Lab1.cpp](#)

```
#include "functions.h"
```

```
int main()
{
    string inputName = "input.txt";
    string outputName = "output.txt";

    int mode = chooseMode();

    createInputFile(inputName, mode);
    createOutputFile(outputName, inputName);

    printFile(inputName, "Input File:");
    printFile(outputName, "\nOutput File:");
}
```

[functions.h](#)

```
#pragma once
```

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

```
using namespace std;
```

```
int chooseMode();
```

```
void createInputFile(string, int);  
void createOutputFile(string, string);  
void printFile(string, string);
```

functions.cpp

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

```
using namespace std;
```

```
enum Mode {  
    OVERWRITING = 1,  
    ADDING = 2  
};
```

```
int chooseMode() {  
    string ch;  
    cout << "Choose writing mode (1 - for overwriting text, 2 - for adding text): ";  
    cin >> ch;  
  
    while (ch != "1" && ch != "2") {  
        cout << "You may only enter a '1' or an '2'!" << endl;  
        cout << "Choose writing mode (1 - for overwriting text, 2 - for adding text): ";  
        cin >> ch;  
    }  
  
    return stoi(ch);  
}
```

```
void createInputFile(string name, int mode) {  
  
    ofstream inputFile;  
    if (Mode::OVERWRITING == mode) {  
        inputFile.open(name);  
    }  
    else {  
        inputFile.open(name, ios::app);  
    }  
  
    string line;  
    size_t combinationCode = 7;
```

```

cout << "Input your text. ENTER for next line. CTRL+G to stop" << endl;
cin.ignore();

if (inputFile) {

    getline(cin, line);
    if (line[0] != combinationCode) {
        inputFile << line;
        getline(cin, line);
        while (line[0] != combinationCode) {
            inputFile << "\n" << line;
            getline(cin, line);
        }
    }
}
inputFile.close();
}

```

```

vector <string> getWords(string name) {
    vector <string> words;
    string word;
    ifstream inputFile(name);

    while (!inputFile.eof()) {
        inputFile >> word;
        words.push_back(word);
    }

    inputFile.close();
    return words;
}

```

```

int countWords(string word, vector <string> words) {
    int counter = 0;
    for (string s : words) {
        if (s == word) {
            ++counter;
        }
    }
    return counter;
}

```

```

bool isNum(string s)
{
    for (char ch : s) {

```

```

        if (!isdigit(ch) and ch != '-') return false;
    }
    return true;
}

int inputNum() {
    string n;
    cout << "Enter number N: ";
    cin >> n;

    while (!isNum(n)) {
        cout << "You can only enter an interger number: ";
        cin >> n;
    }
    return stoi(n);
}

vector <string> removeWord(vector<string> words, string word) {

    vector <string> newWords;

    for (string s : words) {
        if (s != word) {
            newWords.push_back(s);
        }
    }

    return newWords;
}

vector <string> createRepeatingVector(string name, int n) {

    vector <string> words = getWords(name);

    vector <string> updatedWords = words;

    int counter;
    for (string word : words) {
        counter = countWords(word, words);
        if (counter <= n) {
            updatedWords = removeWord(updatedWords, word);
        }
    }

    return updatedWords;
}

```

```

void oddEvenSort(vector <string> &words) {
    bool sorted = false;
    while (!sorted) {
        sorted = true;

        for (int i = 0; i < words.size() - 1; i += 2) {
            if (words[i].length() > words[i + 1].length()) {
                string temp = words[i];
                words[i] = words[i + 1];
                words[i + 1] = temp;
                sorted = false;
            }
        }

        for (int i = 1; i < words.size() - 1; i += 2) {
            if (words[i].length() > words[i + 1].length()) {
                string temp = words[i];
                words[i] = words[i + 1];
                words[i + 1] = temp;
                sorted = false;
            }
        }
    }
}

```

```

void createOutputFile(string outputName, string inputName) {

    int n = inputNum();
    vector <string> repeatingWords = createRepeatingVector(inputName, n);
    oddEvenSort(repeatingWords);

    ofstream outputFile(outputName);

    if (repeatingWords.size() > 0) outputFile << repeatingWords[0];
    for (int i = 1; i < repeatingWords.size(); i++) {
        outputFile << " " << repeatingWords[i];
    }
    outputFile.close();
}

```

```

void printFile(string name, string text) {

    string line;
    ifstream inputFile(name);

```

```

        cout << text << endl;

        while (!inputFile.eof()) {
            getline(inputFile, line);
            cout << line << endl;
        }

        inputFile.close();
    }
}

```

Код Python

main.py

```

import functions as f

inputName = "/Users/oleksandrhumeniuk/Desktop/input.txt"
outputName = "/Users/oleksandrhumeniuk/Desktop/output.txt"

mode = f.chooseMode()

f.createInputFile(inputName, mode)
f.createOutputFile(outputName, inputName)

f.printFile(inputName, "\nInput File:");
f.printFile(outputName, "\n\nOutput File:");

```

functions.py

```

def chooseMode():
    ch = input("Choose writing mode (1 - for overwriting text, 2 - for adding text): ")
    while ch != "1" and ch != "2":
        print("You may only enter a '1' or an '2'!")
        ch = input("Choose writing mode (1 - for overwriting text, 2 - for adding text): ")

    if ch == "1":
        return "w"
    else:
        return "a"

def createInputFile(name, mode):

```

```
with open(name, mode) as inputFile:
    combinationCode = 7
    print("Input your text. ENTER for next line. Empty line to stop")
```

```
    line = input()
    if line != "":
        inputFile.write(line)
        line = input()
        while line != "":
            inputFile.write("\n" + line)
            line = input()
```

```
def inputNum():
    num = input("Enter number N: ")
    while (not num.isdigit()) or int(num) < 0:
        print(f'{num} isn't an integer!')
        num = input("Enter number N:")
    return int(num)
```

```
def getWords(name):
    words = []
    with open(name, "r") as inputFile:
        while True:
            line = inputFile.readline()
            if not line: break
            words.extend(line.split())
    return words
```

```
def countWords(word, words):
    counter = 0
    for s in words:
        if s == word:
            counter += 1
    return counter
```

```
def createRepeatingWords(name, n):
    words = getWords(name)
    updatedWords = words
    for word in words:
        counter = countWords(word, words)
        if counter <= n:
```



```
        updatedWords = [s for s in updatedWords if s != word]
    return updatedWords
```

```
def oddEvenSort(words):
    isSorted = False
    while not isSorted:
        isSorted = True

        for i in range(0, len(words) - 1, 2):
            if len(words[i]) > len(words[i + 1]):
                words[i], words[i+1] = words[i+1], words[i]
            isSorted = False

        for i in range(1, len(words) - 1, 2):
            if len(words[i]) > len(words[i + 1]):
                words[i], words[i+1] = words[i+1], words[i]
            isSorted = False
```

```
def createOutputFile(outputName, inputName):
    n = inputNum()
    repeatingWords = createRepeatingWords(inputName, n)
    oddEvenSort(repeatingWords)

    with open(outputName, "w") as outputFile:
        if len(repeatingWords) > 0: outputFile.write(repeatingWords[0])
        for i in range(1, len(repeatingWords)):
            outputFile.write(" " + repeatingWords[i])
```

```
def printFile(name, text):

    with open(name, "r") as file:
        print(text)
        while True:
            line = file.readline()
            if not line: break
            print(line, end="")
```

Результати виконання коду

```
Python 3.9.7 (v3.9.7:1016ef3790, Aug 30 2021, 16:39:15)
[Clang 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /Users/oleksandrhumeniuk/Desktop/OP_Lab1.py =====
Choose writing mode (1 - for overwriting text, 2 - for adding text): 1
Input your text. ENTER for next line. Empty line to stop
abc abc
ab cde
a a a
abcde ab
abcde
cde
sa kdajsd
adkas jdkvcx

Enter number N:
isn't an integer!
Enter number N:;
; isn't an integer!
Enter number N:1

Input File:
abc abc
ab cde
a a a
abcde ab
abcde
cde
sa kdajsd
adkas jdkvcx

Output File:
a a a ab ab abc abc cde cde abcde abcde
>>>
```

Ln: 35 Col: 4

```
input
abc abc
ab cde
a a a
abcde ab
abcde
cde
sa kdajsd
adkas jdkvcx
```

