

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 27

Виконав студент	<u>ІП-15, Пономаренко Маргарита Альбертівна</u> (шифр, прізвище, ім'я, по батькові)
Перевірів	Вечерковська Анастасія Сергіївна ( прізвище, ім'я, по батькові)

## Лабораторна робота 1

### Текстові файли

#### Індивідуальне завдання

#### Варіант 27

27. Створити текстовий файл. Кожен парний рядок вихідного файлу переписати в перший новий текстовий файл, кожен непарний - у другий. У файлі з парними рядками змінити рядки таким чином, щоб слова кожного рядка були лексично впорядковані за алфавітом. Вивести вміст вихідного і створених файлів.

#### Код

#### C++

Lab-1(2 semester).cpp:

```
#include <iostream>

#include <fstream>

#include <vector>

#include <string>

#include <algorithm>

using namespace std;

string createFile();

vector<string> loadFile(const string& fileName);

vector<string> extract(vector<string> data, int pair = 1);

void saveFile(const vector<string> data, const string& fileName);

void sortLine(vector<string>& data);

void printFile(const string& fileName);
```

## Основи програмування 2. Модульне програмування

```
const string firstFileName = "firstFile.txt";  
const string secondFileName = "secondFile.txt";  
vector<string> split(const string& line, char splitter = ' ');  
string toString(const vector<string>& data);
```

```
int main() {  
    string baseFileName = createFile();  
    vector<string> baseFileData = loadFile(baseFileName);  
    vector<string> firstFileData = extract(baseFileData, 2);  
    sortLine(firstFileData);  
    saveFile(firstFileData, firstFileName);  
    vector<string> secondFileData = extract(baseFileData);  
    saveFile(secondFileData, secondFileName);  
    printFile(baseFileName);  
    printFile(firstFileName);  
    printFile(secondFileName);  
}
```

```
string createFile() //створення основного файлу
```

```
{  
    string fileName;  
    ofstream file;  
    cout << "Enter file name: ";  
    getline(cin, fileName);  
    file.open(fileName);  
    if (!file.is_open()) {  
        cout << "Error creating file";  
        return "";  
    }  
}
```

## Основи програмування 2. Модульне програмування

```
cout << "Enter file data: \n";

string line;

do {

    getline(cin, line);

    if (line != "") {

        file << line + "\n";

    }

} while (line != "");

file.close();

return fileName;

}
```

**vector<string> loadFile(const string& fileName) //завантаження даних з основного файлу, для подальшого використання**

```
{

    string dataLine;

    vector<string> data;

    ifstream file;

    file.open(fileName);

    if (!file.is_open()) {

        cout << "Error opening file";

        return data;

    }

    while (!file.eof()) {

        getline(file, dataLine);

        if (!file.eof()) data.push_back(dataLine);

    }

    return data;

}
```

## Основи програмування 2. Модульне програмування

```
vector<string> extract(vector<string> data, int pair)
{
    vector<string> temp;
    for (size_t i = 0; i < data.size(); i++)
    {
        if (pair == 2) { //виділення парних рядків
            if (i % 2 != 0) {
                temp.push_back(data.at(i));
            }
        }
        else if (pair == 1) { //виділення непарних рядків
            if (i % 2 == 0) {
                temp.push_back(data.at(i));
            }
        }
    }
    return temp;
}

void sortLine(vector<string>& data) //сортування парних рядків
{
    for (size_t i = 0; i < data.size(); i++) {
        auto v = split(data.at(i), ' ');
        sort(v.begin(), v.end());
        data[i] = toString(v);
    }
}
```

```
void saveFile(const vector<string> data, const string& fileName) //збереження даних у відповідний файл
```

```
{  
  
    ofstream file;  
  
    file.open(fileName);  
  
    if (!file.is_open()) {  
        cout << "Error creating file";  
  
        return ;  
    }  
  
    string line;  
  
    for(auto line:data) {  
        file << line + "\n";  
    }  
  
    file.close();  
}
```

**void printFile(const string& fileName) //виведення даних з файлу**

```
{  
  
    vector<string> baseFileData = loadFile(fileName);  
  
    for (auto line : baseFileData) {  
        cout << line + "\n";  
    }  
  
    cout << "\n";  
}
```

**vector<string> split(const string& line, char splitter) //функція, що розділяє рядок по словах**

```
{  
  
    vector<string> temp;  
  
    string tempLine = "";  
  
    if (line.length() == 0) {  
        return temp;  
    }
```

```
}  
  
for (auto ch : line) {  
    if (ch == splitter) {  
        temp.push_back(tempLine);  
        tempLine = "";  
    }  
    else {  
        tempLine += ch;  
    }  
}  
temp.push_back(tempLine);  
return temp;  
}
```

**string toString(const vector<string>& data) //приймає відсортований вектор слів, повертає рядок з'єднаних слів через пробіл**

```
{  
    string tempLine = "";  
    for (auto line : data) {  
        tempLine += line + " ";  
    }  
    tempLine.pop_back();  
    return tempLine;  
}
```

Python

Lab\_1.py:

```
from functions import *
```

## Основи програмування 2. Модульне програмування

```
file_name = create_file()

data = []

if file_name is not None:

    data = load_file(file_name)

data_first = extract(data, 0)

data_second = extract(data, 1)

sort_data = sort_list(data_second)


first_file_name = 'firstFile.txt'

second_file_name = 'secondFile.txt'

save_file(sort_data, first_file_name)

save_file(data_first, second_file_name)


print('Inputed data:')

print_file(file_name)

print('Sorted pair lines:')

print_file(first_file_name)

print('Unpaired lines:')

print_file(second_file_name)
```

**functions.py:**

```
import sys

def create_file(): #створення основного файлу з іменем, яке вводить користувач

    file_name = input('Enter file name: ')

    try:

        file = open(file_name, "w+")

    except IOError:

        print('Error creating file!')

        return None
```



## Основи програмування 2. Модульне програмування

```
print('Press Enter to move to a new line\nPress Ctrl+D to stop data input\nEnter  
file data:')
```

```
line = sys.stdin.read()
```

```
line = line[:len(line)-1]
```

```
file.write(line)
```

```
file.close()
```

```
return file_name
```

```
def load_file(file_name): #вивантаження даних з основного файлу, для їх подальшого  
використання
```

```
data = []
```

```
try:
```

```
    file = open(file_name, "r")
```

```
except IOError:
```

```
    print('Error opening file!')
```

```
    return None
```

```
for line in file:
```

```
    data.append(line.replace('\n', ''))
```

```
return data
```

```
def extract(data, first): #функція, що повертає парні або непарні рядки
```

```
    return data[first::2]
```

```
def sort_list(data): #функція приймає парні рядки, а потім вже з усіх відсортованих  
рядків створює єдинну змінну
```

```
spisok = []
```

```
for line in data:
```

```
    spisok.append(sort_word(line))
```

```
return spisok
```

## Основи програмування 2. Модульне програмування

```
def sort_word(line): #функція приймає парні рядки, розбиваючи їх по слову та додаючи
до списку, а потім об'єднує список в рядкову змінну

    spisok = line.split()

    spisok.sort()

    return ' '.join(spisok)


def save_file(data, file_name): #записує та зберігає інформацію в файлі

    content = ''

    file = open(file_name, "w+")

    for line in data:

        content += line + '\n'

    file.write(content.rstrip('\n'))

    file.close()


def print_file(file_name): #вивантажує та виводить на екран інформацію зі збереженого
файлу

    file_out = open(file_name)

    print(file_out.read())

    print('*' * 10)
```

### Результат виконання програми

**C++:**

```
Консоль отладки Microsoft Visual Studio
Enter file name: base.txt
Enter file data:
1s this is a first sentence
2s second line in a text
3s third string we can see
4s forth pair line to sort

Inputed data:
1s this is a first sentence
2s second line in a text
3s third string we can see
4s forth pair line to sort

Sorted pair lines:
2s a in line second text
4s forth line pair sort to

Unpaired lines:
1s this is a first sentence
3s third string we can see
```

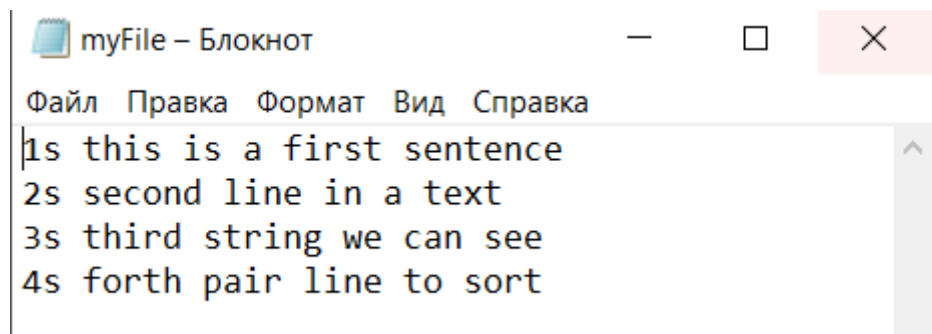
```
base – Блокнот
Файл Правка Формат Вид Справка
1s this is a first sentence
2s second line in a text
3s third string we can see
4s forth pair line to sort
```

```
firstFile – Блокнот
Файл Правка Формат Вид Справка
2s a in line second text
4s forth line pair sort to
```

```
secondFile – Блокнот
Файл Правка Формат Вид Справка
1s this is a first sentence
3s third string we can see
```

**Python:**

```
Enter file name: myFile.txt
Press Enter to move to a new line
Press Ctrl+D to stop data input
Enter file data:
1s this is a first sentence
2s second line in a text
3s third string we can see
4s forth pair line to sort
^D
Inputed data:
1s this is a first sentence
2s second line in a text
3s third string we can see
4s forth pair line to sort
*****
Sorted pair lines:
2s a in line second text
4s forth line pair sort to
*****
Unpaired lines:
1s this is a first sentence
3s third string we can see
*****
```



## Основи програмування 2. Модульне програмування

---

