

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 № 1.1 з дисципліни  
«Основи програмування 2. Модульне програмування»  
«Текстові файли»

Варіант 27

Виконав студент ІП-15, Пономаренко Маргарита Альбертівна  
(шифр, прізвище, ім'я, по батькові)

Перевірів  
( прізвище, ім'я, по батькові)

## Лабораторна робота 1.1

### 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(2term).cpp:

```
OP-Lab1(2term).cpp
OP-Lab1(2term) (Глобальная область)
1 //Створити текстовий файл. Кожен парний рядок вихідного файлу переписати в перший новий текстовий файл,
2 //кожен непарний - у другий. У файлі з парними рядками змінити рядки таким чином,
3 //щоб слова кожного рядка були лексично впорядковані за алфавітом. Вивести вміст вихідного і створених файлів.
4
5 #include <iostream>
6 #include <fstream>
7 #include <vector>
8 #include <string>
9 #include <algorithm>
10
11 using namespace std;
12
13 string createFile();
14 vector<string> loadFile(const string& fileName);
15 vector<string> extract(vector<string> data, int pair = 1);
16 void saveFile(const vector<string> data, const string& fileName);
17 void sortLine(vector<string>& data);
18 void printFile(const string& fileName);
19 const string firstFileName = "firstFile.txt";
20 const string secondFileName = "secondFile.txt";
21 vector<string> split(const string& line, char splitter = ' ');
22 string toString(const vector<string>& data);
23
```

```

24 int main() {
25     string baseFileName = createFile();
26     vector<string> baseFileData = loadFile(baseFileName);
27     vector<string> firstFileData = extract(baseFileData, 2);
28     sortLine(firstFileData);
29     saveFile(firstFileData, firstFileName);
30     vector<string> secondFileData = extract(baseFileData);
31     saveFile(secondFileData, secondFileName);
32
33     cout << "Inputed data:" << endl;
34     printFile(baseFileName);
35     cout << "Sorted pair lines:" << endl;
36     printFile(firstFileName);
37     cout << "Unpaired lines:" << endl;
38     printFile(secondFileName);
39 }
40

```

```

41 string createFile() //створення основного файлу
42 {
43     string fileName;
44     ofstream file;
45     cout << "Enter file name: ";
46     getline(cin, fileName);
47
48     file.open(fileName);
49     if (!file.is_open()) {
50         cout << "Error creating file";
51         return "";
52     }
53
54     cout << "Enter file data: \n";
55     string line;
56     do { //зчитування даних для їх запису у файл
57         getline(cin, line);
58         if (line != "") {
59             file << line + "\n";
60         }
61     } while (line != ""); //зчитування до першого пустого рядка
62     file.close();
63     return fileName;
64 }
65

```

## Основи програмування 2. Модульне програмування

```
66
67 vector<string> loadFile(const string& fileName) //завантаження даних з основного файлу, для подальшого використання
68 {
69     string dataLine;
70     vector<string> data;
71     ifstream file;
72     file.open(fileName);
73     if (!file.is_open()) {
74         cout << "Error opening file";
75         return data;
76     }
77     while (!file.eof()) {
78         getline(file, dataLine);
79         if (!file.eof()) data.push_back(dataLine);
80     }
81     return data;
82 }
83
```

```
84 vector<string> extract(vector<string> data, int pair)
85 {
86     vector<string> temp;
87     for (size_t i = 0; i < data.size(); i++)
88     {
89         if (pair == 2) { //виділення парних рядків
90             if (i % 2 != 0) {
91                 temp.push_back(data.at(i));
92             }
93         }
94         else if (pair == 1) { //виділення непарних рядків
95             if (i % 2 == 0) {
96                 temp.push_back(data.at(i));
97             }
98         }
99     }
100     return temp;
101 }
102
```

## Основи програмування 2. Модульне програмування

```
103 void sortLine(vector<string>& data) //сортування парних рядків
104 {
105     for (size_t i = 0; i < data.size(); i++) {
106         auto v = split(data.at(i), ' ');
107         sort(v.begin(), v.end());
108         data[i] = toString(v);
109     }
110 }
111
112 void saveFile(const vector<string> data, const string& fileName) //збереження даних у відповідний файл
113 {
114     ofstream file;
115     file.open(fileName);
116     if (!file.is_open()) {
117         cout << "Error creating file";
118         return;
119     }
120     string line;
121     for (auto line : data) {
122         file << line + "\n";
123     }
124     file.close();
125 }
126
```

```
127 void printFile(const string& fileName) //виведення даних з файлу
128 {
129     vector<string> baseFileData = loadFile(fileName);
130     for (auto line : baseFileData) {
131         cout << line + "\n";
132     }
133     cout << "\n";
134 }
135
136 vector<string> split(const string& line, char splitter) //виділення слів з рядку
137 {
138     vector<string> temp;
139     string tempLine = "";
140     if (line.length() == 0) {
141         return temp;
142     }
143     for (auto ch : line) {
144         if (ch == splitter) {
145             temp.push_back(tempLine);
146             tempLine = "";
147         }
148         else {
149             tempLine += ch;
150         }
151     }
152     temp.push_back(tempLine);
153     return temp;
154 }
155
```

## Основи програмування 2. Модульне програмування

```
156 string toString(const vector<string>& data) //приймання відсортованого вектору зі словами
157 {
158     string tempLine = "";
159     for (auto line : data) {
160         tempLine += line + " ";
161     }
162     tempLine.pop_back();
163     return tempLine; //повернення слів об'єднаних через пробіл
164 }
```

### Python, Lab\_1.py:

```
Lab_1.py x functions.py x
6 1 # Створити текстовий файл. Кожен парний рядок вихідного файлу переписати в перший новий текстовий файл,
2   # кожен непарний - у другий. У файлі з парними рядками змінити рядки таким чином,
3   # щоб слова кожного рядка були лексично впорядковані за алфавітом. Вивести вміст вихідного і створених файлів.
4   |
5   from functions import *
6   file_name = create_file()
7   data = []
8   if file_name is not None:
9       data = load_file(file_name)
10  data_first = extract(data, 0)
11  data_second = extract(data, 1)
12  sort_data = sort_list(data_second)
13
14  first_file_name = 'firstFile.txt'
15  second_file_name = 'secondFile.txt'
16  save_file(sort_data, first_file_name)
17  save_file(data_first, second_file_name)
18
19  print('Inputed data:')
20  print_file(file_name)
21  print('Sorted pair lines:')
22  print_file(first_file_name)
23  print('Unpaired lines:')
24  print_file(second_file_name)
```

### functions.py:

## Основи програмування 2. Модульне програмування

```
Lab_1.py × functions.py ×
1  import sys
2
3
4  # створення основного файлу з іменем, задане користувачем
5  def create_file():
6      file_name = input('Enter file name: ')
7      try:
8          file = open(file_name, "w+")
9      except IOError:
10         print('Error creating file!')
11         return None
12     print('Press Enter to move to a new line\nPress Ctrl+D to stop data input\nEnter file data:')
13     line = sys.stdin.read()
14     line = line[:len(line)-1]
15     file.write(line)
16     file.close()
17     return file_name
18
19
```

## Основи програмування 2. Модульне програмування

```
20 # вивантаження даних з основного файлу
21 def load_file(file_name):
22     data = []
23     try:
24         file = open(file_name, "r")
25     except IOError:
26         print('Error opening file!')
27         return None
28     for line in file:
29         data.append(line.replace('\n', ' '))
30     return data
31
32 # повернення парних або непарних рядків
33 def extract(data, first):
34     return data[first::2]
35
36
37 # прийняття парних рядків та повернення відсортованих
38 def sort_list(data):
39     spisok = []
40     for line in data:
41         spisok.append(sort_word(line))
42     return spisok
43
44
```

```
46 # прийняття парних рядків, розбивання їх по слову та додавання до списку
47 def sort_word(line):
48     spisok = line.split()
49     spisok.sort()
50     return ' '.join(spisok)
51
52
53 # записування та зберігання інформації в файлі
54 def save_file(data, file_name):
55     content = ''
56     file = open(file_name, "w+")
57     for line in data:
58         content += line + '\n'
59     file.write(content.rstrip('\n'))
60     file.close()
61
62
63 # читання та виведення інформації зі збереженого файлу
64 def print_file(file_name):
65     file_out = open(file_name)
66     print(file_out.read())
67     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
*****
```



