

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.  
Метидології програмування»

«Текстові файли»

Варіант 9

Виконав студент ІП-13 Григоренко Родіон Ярославович  
(шифр, прізвище, ім'я, по батькові)

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

# Лабораторна робота 1

## Тематика

### Варіант 9

9. Створити текстовий файл, рядки якого містять розділені пробілами слова, що складаються із цифр або символів. В кожному рядку, що містить числа, знайти найбільше із них. Переписати такі рядки (що містять числа) у новий текстовий файл наступним чином: починається такий рядок знайденим найбільшим числовим значенням, далі послідовно записуються тільки числові значення відповідного рядка вихідного файлу, розділені комами. Вивести вміст вихідного і створеного файлів.

### Код програми

C++

#### Lab1.cpp

```
#include <iostream>
#include "Header.h"
using namespace std;
int main()
{
    string first_file_name = "first file.txt";
    string second_file_name = "second file.txt";
    inputFile(first_file_name);
    cout << "First file:\n";
    outputFile(first_file_name);
    create_second_file(first_file_name, second_file_name);
    cout << "Second file:\n";
    outputFile(second_file_name);
}
```

#### Header.h

```
#include <fstream>

#include <string>

#include <vector>

using namespace std;

void inputFile(string);

void outputFile(string);

void create_second_file(string, string);

bool isDigit(char);

vector<string> change_line(vector<string> numbers);

vector<string> split(string);
```

## Source.cpp

```
#include "Header.h"

void inputFile(string name) {
    ofstream fileout;
    fileout.open(name);

    if (!fileout.is_open()) {
        cout << "ERROR: Could not open";
    }
    else {
        string line;
        cout << "Enter text (send empty line to finish):\n";
        getline(cin, line);
        while (!line.empty()) {
            fileout << line << "\n";
            getline(cin, line);
        }
    }

    fileout.close();
}

void outputFile(string name)
{
    ifstream filein;
    filein.open(name);

    if (!filein.is_open()) {
        cout << "ERROR: Could not open";
    }
    else {
        string line;
        while (!filein.eof()) {
            getline(filein, line);
            cout << line << "\n";
        }
    }

    filein.close();
}

void create_second_file(string filein_name, string fileout_name)
{
    ifstream filein;
    ofstream fileout;
    filein.open(filein_name);
    fileout.open(fileout_name);

    if (!filein.is_open() || !fileout.is_open()) {
        cout << "ERROR: Could not open";
    }
    else {
        string line;
        while (!filein.eof()) {
            getline(filein, line);
            if (!line.empty()) {
                vector<string> words = split(line);
```

```

vector<string> numbers;
string str = "";
for (int i = 0; i < words.size(); i++) {
    str = "";
    for (int j = 0; j < words[i].size(); j++) {
        if (isDigit(words[i][j])) {
            str += words[i][j];
        }
        else if (isDigit(words[i][j]) == false && str != "") {
            numbers.push_back(str);
            str = "";
        }
    }
    if (str != "") {
        numbers.push_back(str);
    }
}

numbers = change_line(numbers);

for (int i = 0; i < numbers.size(); i++) {
    fileout << numbers[i] << " , ";
}
fileout << "\n";
}
}
}

filein.close();
fileout.close();
}

vector<string> split(string str)
{
    vector<string> res;
    string slice = "";
    str += " ";
    for (int i = 0; i < str.length(); i++) {
        if (str[i] == ' ')
        {
            if (slice.length() > 0) {
                res.push_back(slice);
                slice = "";
            }
        }
        else {
            slice += str[i];
        }
    }
    return res;
}

bool isDigit(char ch) {
    if (ch == '1' || ch == '2' || ch == '3' || ch == '4' || ch == '5' || ch == '6'
        || ch == '7' || ch == '8' || ch == '9' || ch == '0') {
        return true;
    }
    else {
        return false;
    }
}

```

```
}  
}
```

```
vector<string> change_line(vector<string> numbers) {  
    string max = "0";  
    string memory;  
    int index;  
    for (int i = 0; i < numbers.size(); i++) {  
        if (stoi(max) < stoi(numbers[i])) {  
            max = numbers[i];  
            index = i;  
        }  
    }  
    memory = numbers[0];  
    numbers[0] = max;  
    numbers[index] = memory;  
    return numbers;  
}
```









## Тестування:

Microsoft Visual Studio Debug Console

Enter text (send empty line to finish):

```
rviu488f92 894f2 89428hfh9824f9 8f8hiojwjf 02jfuh2489fh
298f98phgfh 98h43gf9gq8-3h 89h3g4q98-g83q489gh 3hq98gh9834g
fijrv 484893 8943f hf4884 hf4930 03f4jf9f 49300
```

First file:

```
rviu488f92 894f2 89428hfh9824f9 8f8hiojwjf 02jfuh2489fh
298f98phgfh 98h43gf9gq8-3h 89h3g4q98-g83q489gh 3hq98gh9834g
fijrv 484893 8943f hf4884 hf4930 03f4jf9f 49300
```

Second file:

```
89428 , 92 , 894 , 2 , 488 , 9824 , 9 , 8 , 8 , 02 , 2489 ,
9834 , 98 , 98 , 43 , 9 , 8 , 3 , 89 , 3 , 4 , 98 , 83 , 489 , 3 , 98 , 298 ,
484893 , 8943 , 4884 , 4930 , 03 , 4 , 9 , 49300 ,
```

## Python

main.py

```
from Functions import *

first_file_name = "first file.txt"
second_file_name = "second file.txt"
input_file(first_file_name)
print("First file:")
output_file(first_file_name)
create_second_file(first_file_name, second_file_name)
print("Second file:")
output_file(second_file_name)
```

Functions.py

```
def input_file(filename):
    with open(filename, 'w') as infile:
        line = input("Enter text (send empty line to finish):
\n")
        while line != "":
            infile.write(line + "\n")
            line = input()
```

```

def output_file(filename):
    with open(filename, 'r') as file:
        lines = file.read().split("\n")
        for i in lines:
            print(i)

def create_second_file(first_file_name, second_file_name):
    with open(first_file_name, 'r') as infile:
        with open(second_file_name, 'w') as outfile:
            lines = infile.read().split("\n")
            for i in range(len(lines)):
                if lines[i] != "":
                    temp = lines[i]
                    words = temp.split()
                    numbers = []
                    st = ""
                    for j in range (len(words)):
                        st = ""
                        for k in range(len(words[j])):
                            if isDigit(words[j][k]):
                                st += words[j][k]
                            elif isDigit(words[j][k]) ==
False and st != "":
                                numbers.append(st)
                                st = ""
                        if st != "":
                            numbers.append(st)

                    numbers = change_line(numbers)
                    for q in range(len(numbers)):
                        outfile.write(numbers[q] + " , ")
                    outfile.write("\n")

def isDigit(ch):
    if ch == '1' or ch == '2' or ch == '3' or ch == '4' or ch
== '5' or ch == '6' or ch == '7' or ch == '8' or ch == '9' or
ch == '0':
        return True
    else:
        return False

def change_line(numbers):
    max = "0"
    memory = ""

```

```
index = 0
for i in range(len(numbers)):
    if int(max) < int(numbers[i]):
        max = numbers[i]
        index = i
memory = numbers[0]
numbers[0] = max
numbers[index] = memory
return numbers
```



Enter text (send empty line to finish):

```
rauuvv84024fjf04 j4023fj 04f02fh8gbk-3b 93bh8320 c,-,5v4uvb7b3 492c
2349p8ghhv932ph 98 h294f8 9h4308fh340[ h[048 h8043h 08fh[02fn1vw9h34
43f0[fh j0f43[0hf024fh nawbgpsdjnbh796p4bn4w8 n5309v3v-j
```

First file:

```
rauuvv84024fjf04 j4023fj 04f02fh8gbk-3b 93bh8320 c,-,5v4uvb7b3 492c
2349p8ghhv932ph 98 h294f8 9h4308fh340[ h[048 h8043h 08fh[02fn1vw9h34
43f0[fh j0f43[0hf024fh nawbgpsdjnbh796p4bn4w8 n5309v3v-j
```

Second file:

```
84024 , 04 , 4023 , 04 , 02 , 8 , 3 , 93 , 8320 , 5 , 4 , 7 , 3 , 492 ,
8043 , 8 , 932 , 98 , 294 , 8 , 9 , 4308 , 340 , 048 , 2349 , 08 , 02 , 9 , 34 ,
5309 , 0 , 0 , 43 , 0 , 024 , 796 , 4 , 4 , 8 , 43 , 3 ,
```

## Тестування:

## Висновки:

Я вивчив особливості створення і обробки текстових файлів даних. Застосував ці навички на практиці.