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. Об’єктно-орієнтоване програмування»

«Файли даних. Текстові файли»

Варіант 21

Виконав студент ІП-11, Ляля Іван Олександрович

Перевірила Вітковська Ірина Іванівна\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Київ 2022

**Лабораторна робота 1**

**Файли даних. Текстові файли**

**Мета** – вивчити особливості створення і обробки текстових файлів даних.

**Задача:**

Створити текстовий файл. Переписати його вміст у новий текстовий файл, замінюючи кожен рядок символом '-', якщо він у вихідному файлі закінчується на літеру, відмінну від символа пропуску, коми або крапки. Вивести вміст вихідного і створеного файлів.

**Постановка задачі:**

На С++ ми підключаємо бібліотеки iostream для потокового i/o, fstream для файлового i/o, string роботи з текстом через рядки і їх методи. На обох мовах опишемо функції File\_Create, ReWrite, Output, які прийматимуть параметри-імена файлів.

File\_Create: буде запитувати користувача, чи хоче він дописати чи переписати файл і відповідно до цього відкриватиме файл у режимі append або write. Підказуємо користувачу вводити файл і натиснути комбінацію Ctrl+E щоб закінчити ввід. Створимо булеву змінну flag, що буде набувати значення 0, коли натиснуто цю комбінацію клавіш. Коли вона буде значення 1 текст буде просто записуватись у файл через змінну з перевіркою чи введений рядок не пустий; коли 0 – усі символи від Ctrl+E включно будуть стерті і також записані у файл. Закриваємо файл.

ReWrite: відкриває вхідний файл у режимі зчитування з файлу і вихідний у режимі запису в файл. Поки не кінець файлу, відбувається зчитування рядка і, якщо його довжина більше 0, то відбувається такий блок: якщо останній символ рядка пробіл, крапка, або кома, то у файл записується рядок, інакше – символ ‘-’. Закриваємо файли.

Output: відкриває файл в режимі зчитування з файлу, і поки не кінець файлу, функція по рядку виводить його вміст; закриваємо файл.

Головна функція є такою: присвоюємо імена: вхідному файлу “source file.txt”, вихідному -”created file.txt”. Викликаємо функцію заповнення першого файлу, викликаємо функцію обробки тексту першого файлу і переписування цього тексту в другий файл, виводимо вміст вхідного файлу, виводимо вміст вихідного файлу.

**Текст і результат роботи програми на С++:**

**cpp lab 1.cpp:**

#include "Header.h"

int main()

{

string name1 = "source file.txt", name2 = "created file.txt";

File\_Create(name1);

ReWrite(name1, name2);

cout << "\nFile 1 content:" << endl;

Output(name1);

cout << "\nFile 2 content:" << endl;

Output(name2);

return 0;

}

**Header.h:**

#pragma once

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

void File\_Create(string);

void ReWrite(string, string);

void Output(string);

**Source.cpp:**

#include "Header.h"

void File\_Create(string name)

{

bool mode;

cout << "Do you want to append file (1) or create a clear file (0)? ";

cin >> mode;

ofstream outfile1;

if (mode) outfile1.open(name, ios::app);

else outfile1.open(name);

cout << "Input text to the file. Press Ctrl+E to stop inputing\n";

string str;

bool flag = true;

while (flag)

{

getline(cin, str);

for (int i = 0; i < str.size(); i++)

{

if (str[i] == 5)

{

flag = false;

str.erase(i);

}

}

if (str.size() > 0) outfile1 << str << endl;

}

outfile1.close();

}

void ReWrite(string name1, string name2)

{

ifstream infile1(name1);

ofstream file2(name2);

string str;

while (!infile1.eof())

{

getline(infile1, str);

if (str.size() > 0)

{

if (str[str.size() - 1] == ' ' || str[str.size() - 1] == ',' || str[str.size() - 1] == '.') file2 << str << endl;

else file2 << '-' << endl;

}

}

infile1.close();

file2.close();

}

void Output(string name)

{

ifstream file(name);

string str;

while (!file.eof())

{

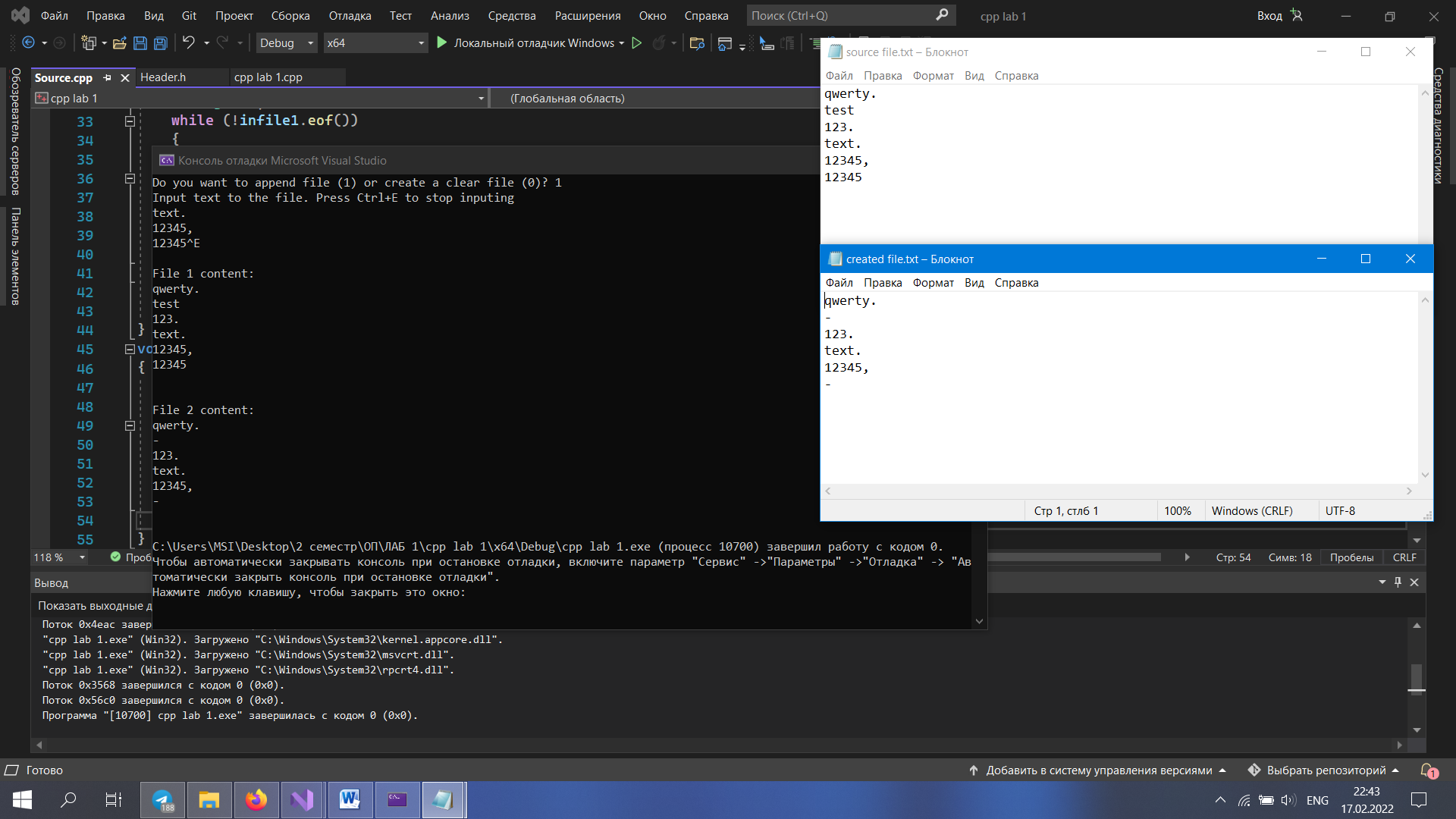
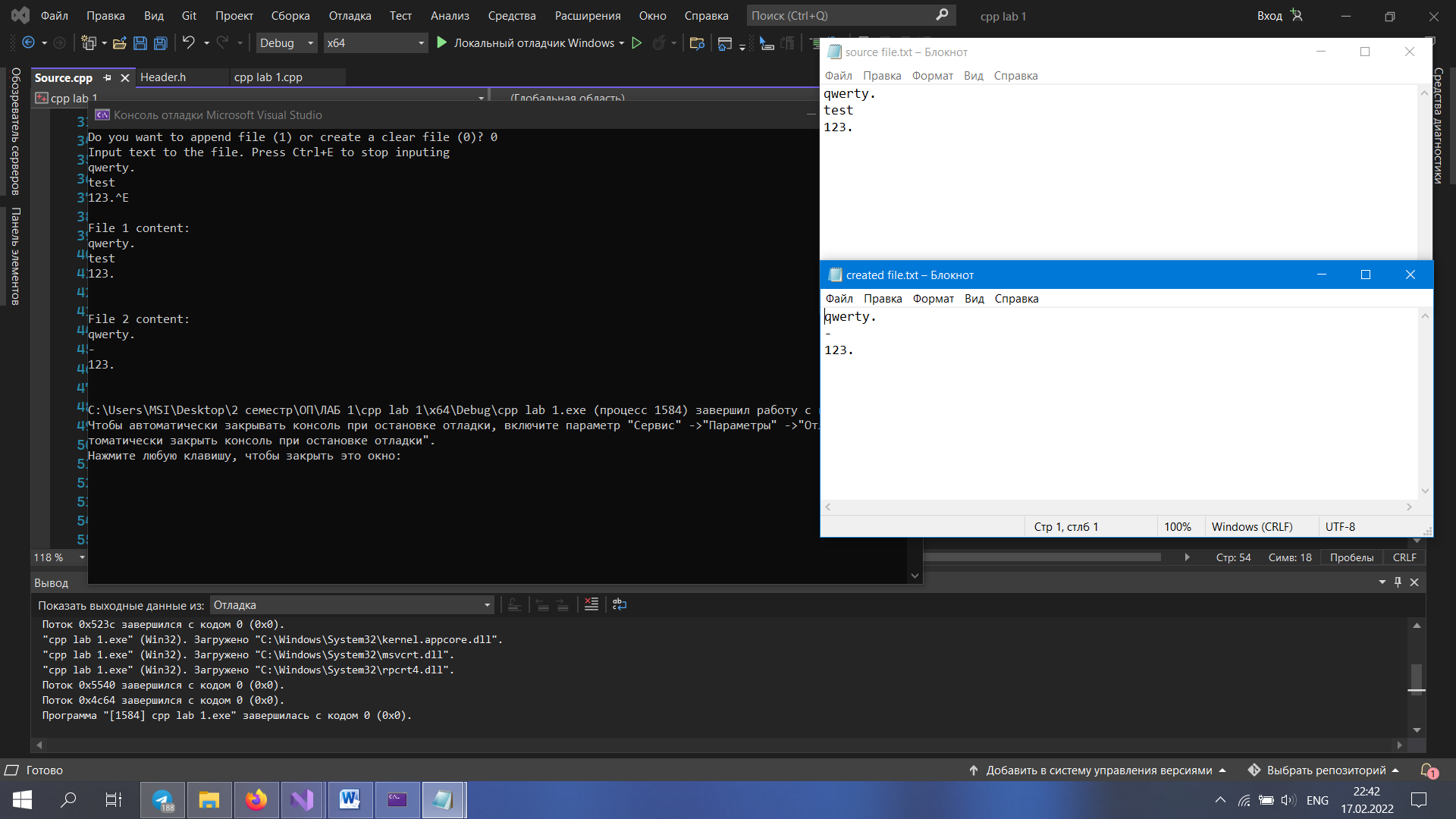
getline(file, str);

cout << str << endl;

}

file.close();

}



**Текст і результат роботи програми на Python:**

**python\_lab\_1.py:**

import module

name1="source file.txt"

name2="created file.txt"

module.File\_Create(name1)

module.ReWrite(name1,name2)

print("File 1 content:")

module.Output(name1)

print("File 2 content:")

module.Output(name2)

**module.py:**

def File\_Create(name):

mode=int(input("Do you want to append file (1) or create a clear file (0)? "))

if mode==1: m='a'

else: m='w'

with open(name,m) as outfile1:

print("Input text to the file. Press Ctrl+E to stop inputing\n")

flag=True

while flag:

string=input()

for i in range(len(string)):

if string[i]==chr(5):

flag=False

string=string[:i:]

if len(string)>0:

outfile1.write(string+'\n')

def ReWrite(name1,name2):

with open(name1) as infile1:

with open(name2,'w') as file2:

for line in infile1:

string=line

if len(string)>0:

if string[-2]==' ' or string[-2]=='.' or string[-2]==',':

file2.write(string)

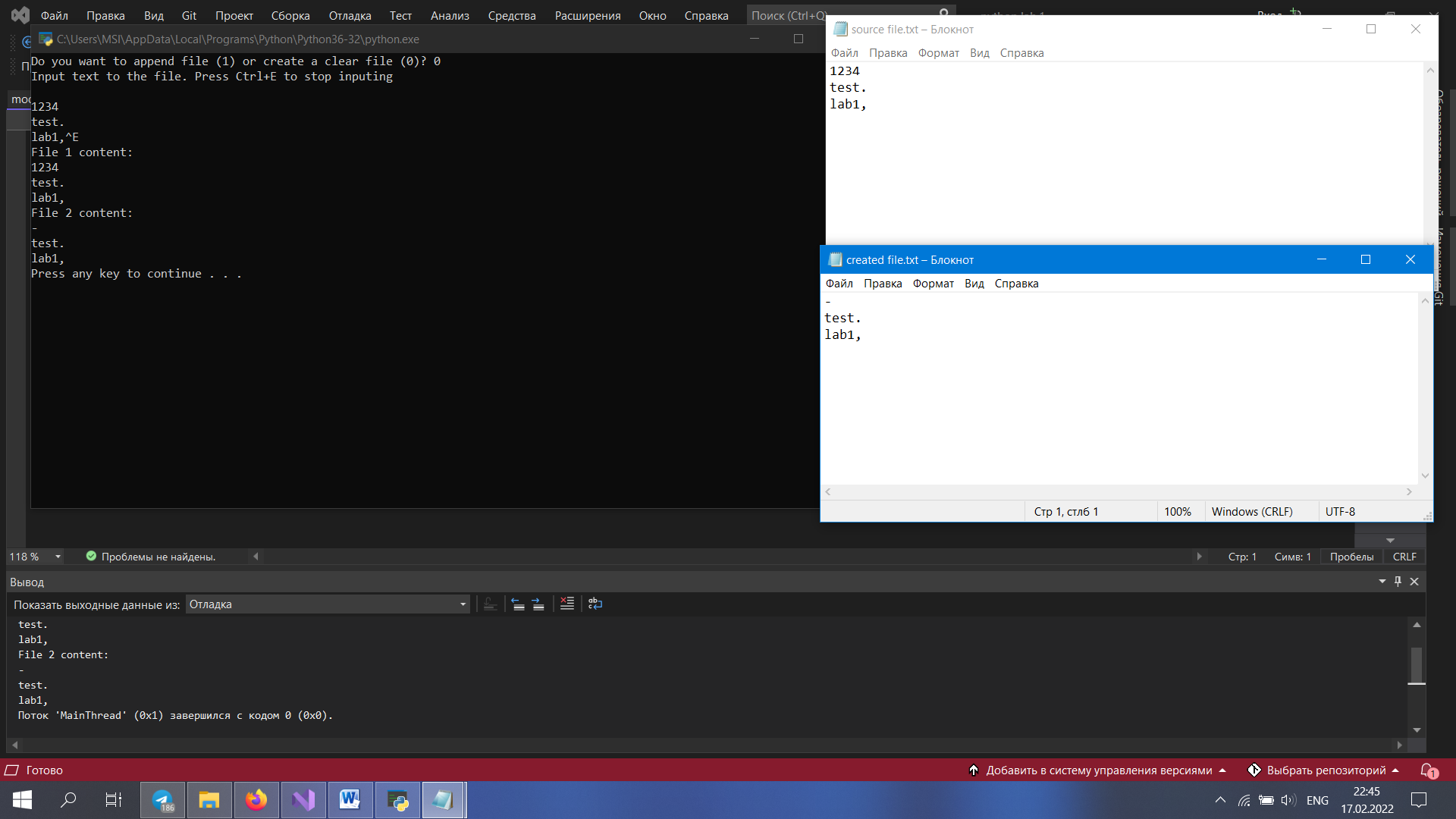
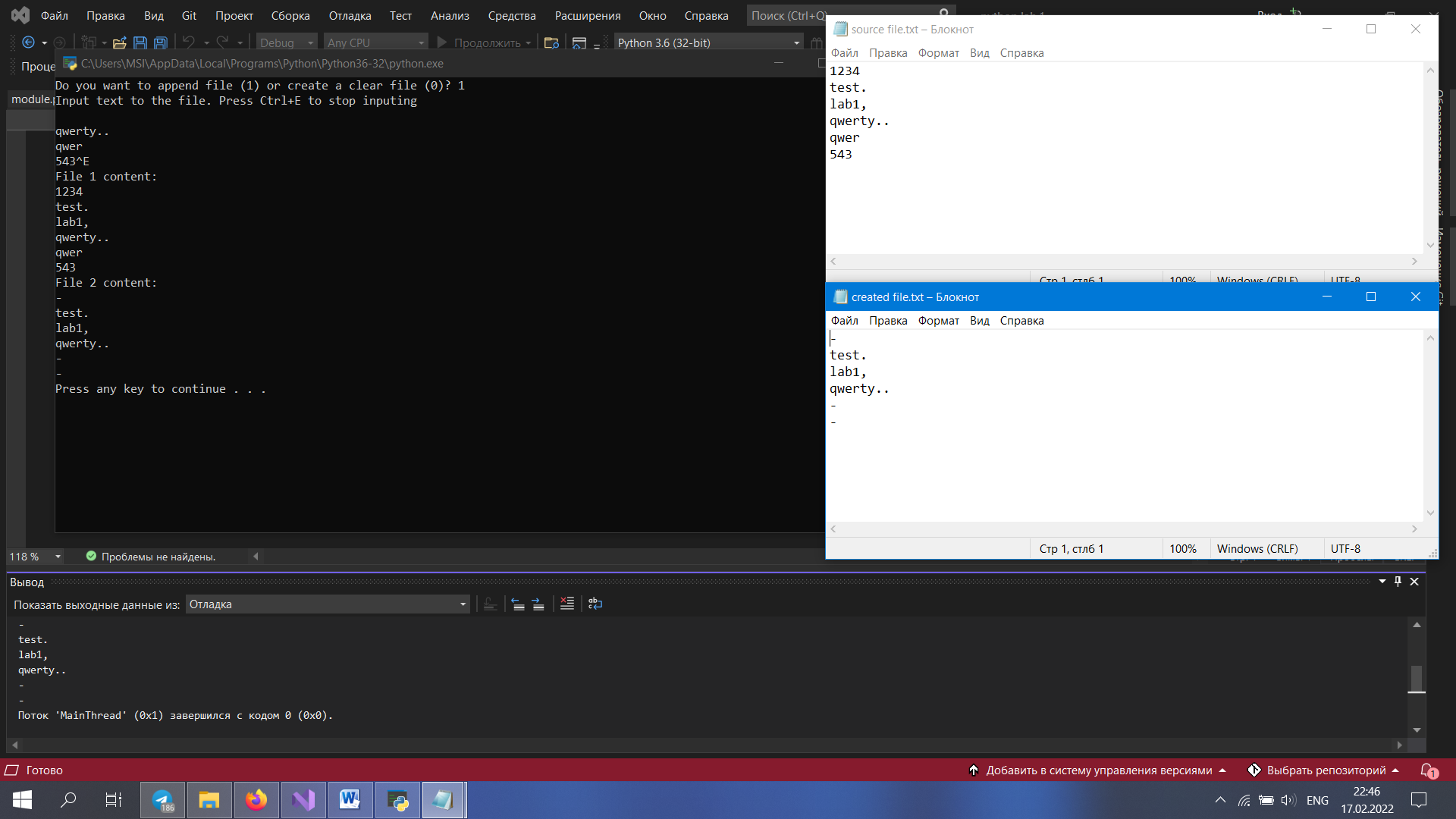
else: file2.write('-\n')

def Output(name):

with open(name) as filee:

for line in filee:

print(line, end = "")

**Висновок:**

Під час виконання лабораторної роботи мені довелося освоїти навички роботи з файлами даних, а саме текстовими файлами. На мові С++ це реалізується методами бібліотеки fstream. Власне обробка тексту відбувалась уже вивченими раніше методами роботи з рядками. Особливістю виконання роботи стала необхідність організувати ввід до певної комбінації клавіш, проте це реалізується простою перевіркою символів введеного рядка на наявність символа з певним кодом. Структурне програмування при роботі з файлами реалізується простіше, оскільки в функції в основному потрібно передавати лише назву файла, з яким буде відбуватись подальша робота.