# 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»

«Файли даних. Текстові файли»

Варіант 29

Виконав студент ІП-11 Тарасьонок Дмитро Євгенович

(шифр, прізвище, ім'я, по батькові)

Перевірив Вітковська Ірина Іванівна

( прізвище, ім'я, по батькові)

Київ 2022

Мета: вивчити особливості створення і обробки текстових файлів

Умова задачі: Створити два текстових файли. Переписати в новий текстовий файл рядки, які є в першому вихідному файлі й відсутні у другому. Визначити кількість таких рядків. Вивести вміст вихідних і створених файлів.

Постановка задачі: Для виконання поставленої задачі треба спроектувати функцію введення рядків, при кожному введенні рядка буде шукатися символ 0x7, що відповідає комбінації клавіш Ctrl+G, видалятиме все, що знаходиться після цього символу та додавати цей рядок до вектору, в якому будуть зберігатися всі рядки. Таку операцію виконати двічі. Далі треба пройтися по кожному елементу першого вектора й перевірити, чи немає цього елементу в другому векторі. Якщо немає, треба перевірити, чи цей елемент перевірявся раніше (елементам у векторі необов’язково бути унікальними) і, якщо не перевірявся, додати цей рядок у третій вектор, який потім записати у файл, який після цього треба прочитати й вивести всі результати на екран. Кількість унікальних рядків же буде дорівнювати розміру вектора.

Програма мовою C++:

Файл main.cpp

/\*

\* 1 курс, 2 семестр, Основи програмування, лабораторна робота №1.

\*

\* Варіант 29

\* Створити два текстових файли. Переписати в новий текстовий файл рядки,

\* які є в першому вихідному файлі й відсутні у другому. Визначити кількість

\* таких рядків. Вивести вміст вихідних і створених файлів.

\*/

#include <locale>

#include <clocale>

#include <Windows.h>

#include <algorithm>

#include "Functions.h"

using namespace std;

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

locale::global(locale("rus\_rus.1251"));

string sFirstFileName = "file1.txt";

string sSecondFileName = "file2.txt";

string sThirdFileName = "file3.txt";

char ch;

cout << "Ви бажаєте перезаписати перший файл? [y/n]: ";

cin >> ch;

bool bTruncateFirst = ch == 'y' || ch == 'Y';

cout << "Вводьте перший файл (Ctrl + G для завершення вводу):" << endl;

writeFile(sFirstFileName, bTruncateFirst, true);

cout << "Ви бажаєте перезаписати другий файл? [y/n]: ";

cin >> ch;

bool bTruncateSecond = ch == 'y' || ch == 'Y';

cout << "Вводьте другий файл (Ctrl + G для завершення вводу):" << endl;

writeFile(sSecondFileName, bTruncateSecond, true);

vector<string> vFirstFileLines = readFile(sFirstFileName);

vector<string> vSecondFileLines = readFile(sSecondFileName);

vector<string> vUniqueLines = findUniqueElements(vFirstFileLines, vSecondFileLines);

ofstream fThirdFile{ sThirdFileName };

outputVector(vUniqueLines, false, fThirdFile);

fThirdFile.close();

vector<string> vThirdFileLines = readFile(sThirdFileName);

cout << endl << "Перший файл:" << endl;

outputVector(vFirstFileLines);

cout << endl << "Другий файл:" << endl;

outputVector(vSecondFileLines);

cout << endl << "Третій файл (унікальні елементи першого файлу):" << endl;

outputVector(vThirdFileLines);

cout << endl << "Кількість таких елементів: " << vUniqueLines.size() << endl;

}

Файл Function.h

#pragma once

#include <vector>

#include <iostream>

#include <string>

#include <fstream>

using namespace std;

void writeFile(const string& sName, bool bTruncate, bool bIgnoreCin);

vector<string> readFile(const string& name);

void outputVector(const vector<string>& v, bool bEndWithNewline = true, ostream& os = cout);

vector<string> findUniqueElements(const vector<string>& vFirstLines, const vector<string>& vSecondLines);

Файл Functions.cpp

#include "Functions.h"

void writeFile(const string& sName, bool bTruncate, bool bIgnoreCin)

{

ofstream ost{ sName, bTruncate ? ios\_base::out : ios\_base::app };

ost.imbue(locale("rus\_rus.1251"));

bool bBreak = false;

if (bIgnoreCin)

{

cin.ignore();

}

while (!bBreak)

{

string sLine = "";

getline(cin, sLine);

int nBreakPos = sLine.find(0x7);

sLine = sLine.substr(0, nBreakPos);

if (sLine.length() != 0)

{

ost << sLine << endl;

}

if (nBreakPos != -1)

{

bBreak = true;

}

}

}

vector<string> readFile(const string& name)

{

vector<string> vLines;

ifstream ist{ name };

string sLine;

while (!ist.eof())

{

getline(ist, sLine);

if (sLine.length() != 0)

{

vLines.push\_back(sLine);

}

}

return vLines;

}

void outputVector(const vector<string>& v, bool bEndWithNewline, ostream& os)

{

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

{

os << v[i];

if (bEndWithNewline || i != v.size() - 1)

{

os << endl;

}

}

}

vector<string> findUniqueElements(const vector<string>& vFirstLines, const vector<string>& vSecondLines)

{

vector<string> vUniqueElements;

for (const string& sLine : vFirstLines)

{

auto it = find(vSecondLines.begin(), vSecondLines.end(), sLine);

if (it == vSecondLines.end() && find(vUniqueElements.begin(), vUniqueElements.end(), sLine) == vUniqueElements.end())

{

vUniqueElements.push\_back(sLine);

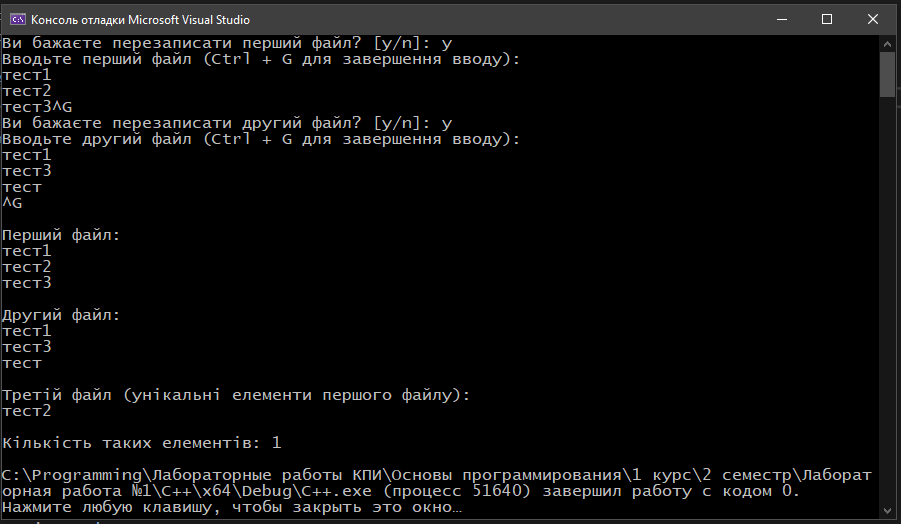
}

}

return vUniqueElements;

}

Виконання коду мовою C++:



Програма мовою Python

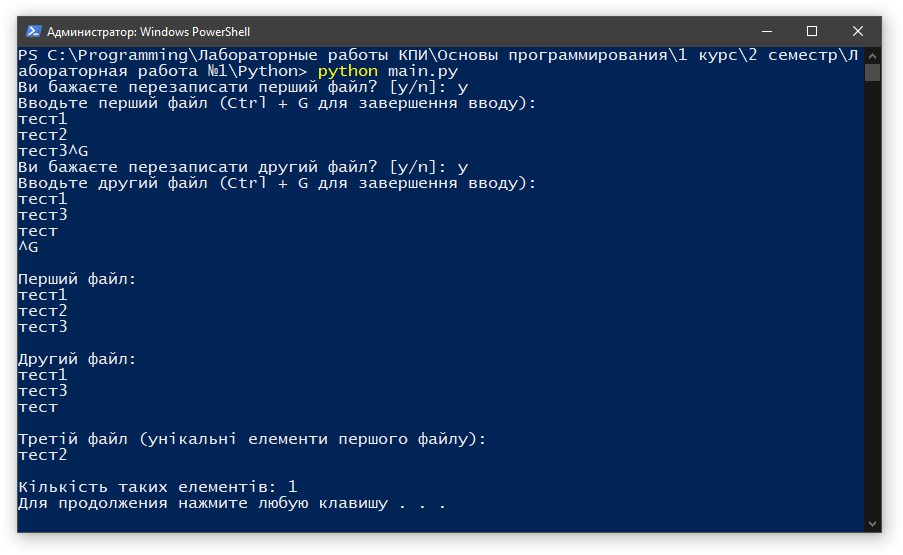
Файл main.py

*"""  
1 курс, 2 семестр, Основи програмування, лабораторна робота №1.  
  
Варіант 29  
Створити два текстових файли. Переписати в новий текстовий файл рядки,  
які є в першому вихідному файлі й відсутні у другому. Визначити кількість  
таких рядків. Вивести вміст вихідних і створених файлів.  
"""  
from* Functions *import* \*  
  
*if* \_\_name\_\_ == '\_\_main\_\_':  
 first\_file\_name = 'file1.txt'  
 second\_file\_name = 'file2.txt'  
 third\_file\_name = 'file3.txt'  
  
 ch = input('Ви бажаєте перезаписати перший файл? [y/n]: ')  
 truncate\_first = ch.lower() == 'y'  
 print('Вводьте перший файл (Ctrl + G для завершення вводу):')  
 write\_file(first\_file\_name, truncate\_first)  
  
 ch = input('Ви бажаєте перезаписати другий файл? [y/n]: ')  
 truncate\_second = ch.lower() == 'y'  
 print('Вводьте другий файл (Ctrl + G для завершення вводу):')  
 write\_file(second\_file\_name, truncate\_second)  
  
 first\_file\_lines = read\_file(first\_file\_name)  
 second\_file\_lines = read\_file(second\_file\_name)  
 unique\_lines = find\_unique\_elements(first\_file\_lines, second\_file\_lines)  
  
 write\_lines\_to\_file(third\_file\_name, unique\_lines)  
  
 third\_file\_lines = read\_file(third\_file\_name)  
  
 print('\nПерший файл:', '\n'.join(first\_file\_lines), sep='\n')  
 print('\nДругий файл:', '\n'.join(second\_file\_lines), sep='\n')  
 print(  
 '\nТретій файл (унікальні елементи першого файлу):',  
 '\n'.join(third\_file\_lines),  
 sep='\n'  
 )  
 print(f'\nКількість таких елементів: {len(unique\_lines)}')  
  
 system('pause')

Файл Functions/\_\_init\_\_.py

*from* os *import* system  
*import* typing  
  
  
*def* write\_file(name: str, truncate: bool):  
 *"""  
 Здійснює зчитування рядків з клавіатури та записує їх у файл.  
  
 Args:  
 name (str): Назва файлу  
 truncate (bool): Перезаписати файл чи дописати в кінець  
 """  
 with* open(name, 'w' *if* truncate *else* 'a') *as* output\_file:  
 break\_input = *False  
  
 while not* break\_input:  
 line = input()  
  
 break\_pos = line.find(chr(0x7))  
 *# break\_pos = line.find('~')  
  
 if* break\_pos != -1:  
 line = line[:break\_pos]  
 break\_input = *True  
  
 if* line:  
 output\_file.write(line)  
  
 *if* break\_pos == -1:  
 output\_file.write('\n')  
  
  
*def* read\_file(name: str):  
 *"""  
 Зчитує рядки з файлу.  
  
 Args:  
 name (str): Назва файлу  
  
 Returns:  
 lines (typin.List[str])  
 """  
 with* open(name) *as* input\_file:  
 lines = [line.strip() *for* line *in* input\_file.readlines()]  
  
 *return* lines  
  
  
*def* find\_unique\_elements(  
 first\_list: typing.List[typing.Any],  
 second\_list: typing.List[typing.Any]  
):  
 *"""  
 Знаходить значення, що містяться в першому списку, але відсутні в другому.  
  
 Args:  
 first\_list (typing.List[typing.Any]): Перший список  
 second\_list (typing.List[typing.Any]): Другий список  
  
 Returns:  
 (typing.List[typing.Any]): Унікальні елементи списку  
 """  
 return* list(set(first\_list) - set(second\_list))  
  
  
*def* write\_lines\_to\_file(name: str, lines: typing.List[str]):  
 *"""  
 Записує рядки до файлу.  
  
 Args:  
 name (str): Назва файлу  
 lines (typing.List[str]): Рядки, які треба записати  
 """  
 with* open(name, 'w') *as* output\_file:  
 output\_file.write('\n'.join(lines))

Виконання програми мовою Python:



Висновок: під час виконання даної лабораторної роботи я удосконалив теоретичні знання та отримав практичні навички з обробки текстових файлів. Важливим набуттям для себе вважаю те, що я навчився зупиняти введення комбінацією клавіш (у моєму випадку Ctrl+G), що робить програму значно красивішою. Також я вивчив режимі відкриття файлів: ios\_base::out, ios\_base::in, ios\_base::trunc та ін. й навчився використовувати все це на практиці.