# 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.

Базові конструкції»

«Файли Даних»

Варіант 20

Виконав студент Лошак Віктор Іванович

(шифр, прізвище, ім'я, по батькові)

Перевірив Вітковська Ірина Іванівна

( прізвище, ім'я, по батькові)

Київ 2022

**Лабораторна робота 1**

**Мета:**

Вивчити особливості створення і обробки текстових файлів даних.

**Тема:**

Створити текстовий файл. Переписати до нового файлу всі компоненти вихідного файлу, замінивши в них символ 0 на 1 і навпаки. Вивести вміст вихідного і створеного файлів. **Математична модель:**

Для знаходження вмісту результуючого файлу використаємо підпрограми що реалізуватимуть відповідно: читання вмісту вже записаного файлу , зчитування користувацького багаторядкового вводу з наступним записом у файл, інверсія заданих символів у файлі. Для реалізації заданих функцій використаємо стандартні доступні бібліотеки та модулі. Для реалізації інверсії одиниць та нулів використаємо проходження по рядках файлу з наступним пошуку в рядку відповідних символів і їх заміни.

**1)Приклад коду на С++:**

**Func.cpp:**

#include "Func.h"

void printFile(const string& fileName)

{

cout << "The contents of the file " << fileName.substr(fileName.find\_last\_of('/', fileName.length( )) + 1) << ":\n";

std::ifstream fileIn(fileName);

string temp;

while (!fileIn.eof( ))

{

getline(fileIn, temp);

cout << temp << endl;

}

fileIn.close( );

}

void appendFile(const string& fileName)

{

cout << "You are editing: " << fileName.substr(fileName.find\_last\_of('/', fileName.length( )) + 1) << "\n";

cout << "Press ctrl+E to stop the edit. \nPlace your input here:\n";

string temp;

ofstream file(fileName, ios::out | ios::app);

if (file.is\_open( ))

{

getline(cin, temp, char(5));

if(temp.back()!='\n') temp += '\n';

file << temp;

}

else

{

cout << "Error occured while trying to open file" << fileName.substr(fileName.find\_last\_of('/', fileName.length( )) + 1);

}

cout << endl;

file.close( );

}

void rewriteFile(const string& fileName)

{

ofstream file(fileName);

file.close( );

appendFile(fileName);

}

void invertBinaryDigits(const string& fileName, const string& newFileName)

{

ifstream file(fileName);

ofstream newFile(newFileName);

string temp;

while (!file.eof( ))

{

getline(file, temp);

temp += '\n';

for (int i = 0; i < temp.length( ); i++)

{

if (temp[i] == '0') temp.replace(i, 1, "1");

else if (temp[i] == '1') temp.replace(i, 1, "0");

}

newFile << temp;

}

file.close( );

newFile.close( );

}

**Func.h:**

#pragma once

#include <iostream>

#include <fstream>

#include <ostream>

#include <istream>

#include <Windows.h>

#include <thread>

#include <string>

using namespace std;

void rewriteFile(const string& fileName);

void printFile(const string & fileName);

void appendFile(const string& fileName);

void invertBinaryDigits(const string& fileName, const string& newFileName);

**Source.cpp:**

#include "Func.h"

int main( )

{

string path = "C:/Users/User/source/repos/KPI/OP/2Semester/Lab1/Files/";

string firstName = "first\_file.txt";

string secondName = "second\_file.txt";

rewriteFile(path+firstName);

invertBinaryDigits(path + firstName, path + secondName);

printFile(path+firstName);

printFile(path + secondName);

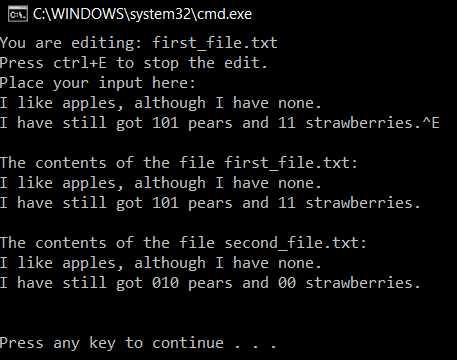
return 0;

}

**I like apples, although I have none.**

**I have still got 101 pears and 11 strawberries.**

**Виконання коду на С++:**

****

**2)Приклад коду на Рython:**

**Lab1\_python\_:**

from module1 import \*

path = "C:/Users/User/source/repos/KPI/OP/2Semester/Lab1/Files/"

firstName = "first\_file.txt"

secondName = "second\_file.txt"

inputText(path+firstName)

InvertBinaryDigits(path+firstName, path+secondName)

printContents(path+firstName)

printContents(path+secondName)

**module1:**

#запис у файл тексту з перевіркою на комбінацію клавіш

def ReadMultipleInp(str,file):

i=0

while i<len(str):

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

str=str[:i:]

if(len(str)!=0):

file.write(str+'\n')

return 0

i+=1

file.write(str+'\n')

return 1

def appendFile(name):

print("Press ctrl+E to stop the edit. \nPlace your input here:")

cont=1

file = open(name,"a")

while(cont):

str\_text=input()

cont = ReadMultipleInp(str\_text,file)

file.close()

def rewriteFile(name):

open(name, "w").close()

appendFile(name)

#читання файлу

def printContents(name):

print("\nread file",name[name.rfind('/')+1:],":")

file=open(name,"r")

for i in file.readlines():

print(i, end=' ')

def InvertBinaryDigits(fileName, newFileName):

streamReader = open(fileName)

streamWriter = open(newFileName, "w")

for i in streamReader.readlines():

index=0

for j in i:

if j == '0':

i=i[:index]+'1'+i[index+1:]

elif j== '1':

i=i[:index]+'0'+i[index+1:]

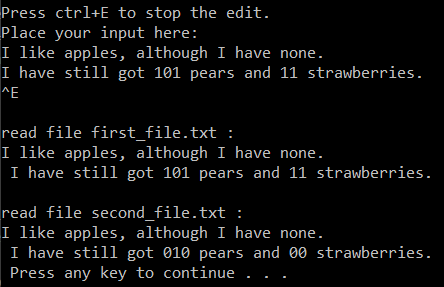
index+=1

streamWriter.write(i)

streamReader.close()

streamWriter.close()

**Виконання коду на python:**

****

**3)Приклад коду на C#**

**Program.cs**

using System;

namespace Lab1\_sharp\_

{

class Program

{

static void Main(string[] args)

{

string path = "C:/Users/User/source/repos/KPI/OP/2Semester/Lab1/Files/";

string firstName = "first\_file.txt";

string secondName = "second\_file.txt";

FileManager.RewriteFile(path + firstName);

FileManager.InvertBinaryDigits(path + firstName, path + secondName);

FileManager.PrintContents(path + firstName);

FileManager.PrintContents(path + secondName);

}

}

}

**Class1.cs**

using System;

using System.Collections.Generic;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Lab1\_sharp\_

{

internal static class FileManager

{

public static void PrintContents(in string fileName)

{

Console.WriteLine($"You are viewing contents of {fileName.Substring(fileName.LastIndexOf('/') + 1)} ");

StreamReader streamReader = new StreamReader(fileName);

Console.WriteLine(streamReader.ReadToEnd());

streamReader.Close();

}

public static void AppendFile(in string fileName)

{

Console.WriteLine($"You are editing {fileName.Substring(fileName.LastIndexOf('/') + 1)} \n " +

$"Press ctrl+E to stop the edit. \nPlace your input here:");

StreamWriter streamWriter=new StreamWriter(fileName, true);

string[] res = ReadMultilineInp();

for (int i = 0; i < res.Length; i++)

{

streamWriter.WriteLine(res[i]);

}

Console.WriteLine('\n');

streamWriter.Close();

}

public static void RewriteFile(in string fileName)

{

StreamWriter streamWriter = new StreamWriter(fileName, false);

streamWriter.Close();

AppendFile(fileName);

}

public static void InvertBinaryDigits(in string fileName, in string newFileName)

{

StreamReader streamReader = new StreamReader(fileName);

StreamWriter streamWriter = new StreamWriter(newFileName, false);

string temp;

while(!streamReader.EndOfStream)

{

temp = streamReader.ReadLine();

for (int i = 0; i < temp.Length; i++)

{

if (temp[i] == '0')

{

char[] letters = temp.ToCharArray();

letters[i] = '1';

temp = new string(letters);

}

else if (temp[i] == '1')

{

char[] letters = temp.ToCharArray();

letters[i] = '0';

temp = new string(letters);

}

}

streamWriter.WriteLine(temp);

}

streamReader.Close();

streamWriter.Close();

}

public static string[] ReadMultilineInp()

{

ConsoleKeyInfo key;

bool contInp;

string temp = "";

do

{

contInp = true;

key = Console.ReadKey();

switch (key.Key)

{

case ConsoleKey.Enter:

temp += '\n';

Console.CursorTop++;

break;

case ConsoleKey.E when key.Modifiers == ConsoleModifiers.Control:

contInp = false;

break;

case ConsoleKey.Backspace:

Console.Write(' ');

Console.CursorLeft--;

break;

default:

temp += key.KeyChar.ToString();

break;

}

}

while (contInp);

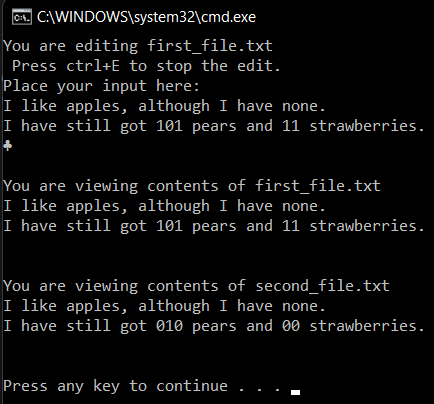
return temp.Split('\n');

}

}

}

**Виконання коду на C#:**



**Висновок:**

Отже за допомогою даного алгоритму ми успішно створили два файли програмним шляхом та записали в нього довільні ввідні дані а також форматований ввід, навчилися читати та обробляти вміст текстових файлів та вивчили роботу потоків, їх типів та модифікаторів , за допомогою використання підпрограм на компільованій мові С++ та python .