Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра інформаційних систем та мереж

### ЗВІТ

про виконання лабораторної роботи №2

на тему: «Oперації над стрічками»

з предмету «Алгоритми та структури даних»

Виконала:

ст. гр. СА-32

Малинка А.А.

Прийняв:

доцент

Щербак С.С.

Львів-2017

**Мета роботи:** набуття практичних навичок застосування операцій над

стрічками.

Лабораторне завдання: (варіант 19)

|  |  |
| --- | --- |
| StrLWord(s,k) | Визначити кількість слів довжиною **к** символів в стрічці **s** |

1.Реалізувати додаток, який дозволяє зчитувати вихідні рядки з файлів та зберігати результати в файл та розробити процедури та функції які забезпечують виконання операції вказаних в завданні.

2. В контрольному прикладі передбачити всі можливі комбінації вхідних параметрів (нульова довжина, вихід за межі стрічки, розмір текстового файлу більше 100 МБ і т.п.), в тому числі і неправильні.

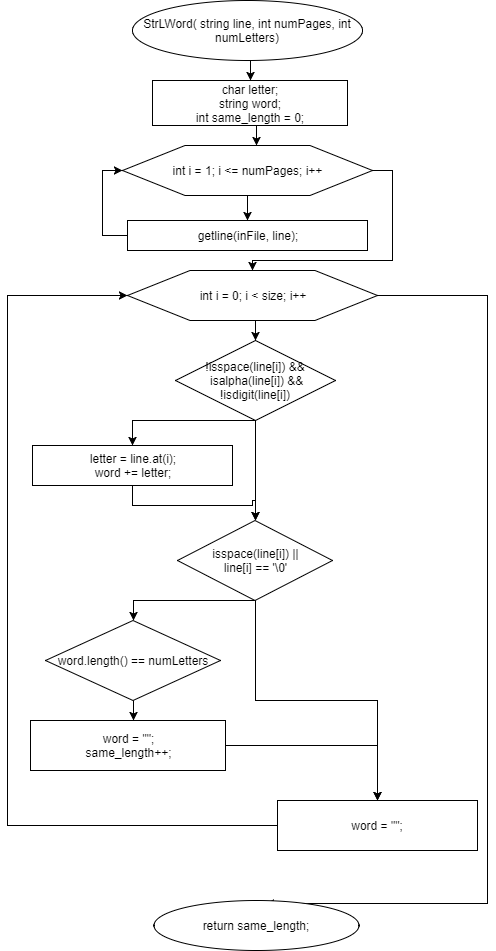
Блок-схема реалізації функції, що підраховує кількість слів довжини к у певній стрічці файлу

Рис 1. Блок-схема функції, що підраховує кількість слів довжини к у певній стрічці файлу

Реалізований алгоритм прочитує заданий файл у «пошуках» стрічки, потім розбиває її на слова, орієнтуючись на пробіли, та не беручи до уваги розділові знаки та числа. Обраховує кількість букв у кожному слові стрічки, порівнює з довжиною яку задає користувач, та підраховує кількість слів довжина яких відповідає.

Код програми ( мовою с++):

/// FindSameLengthWords.h

#include <fstream>

#include <string>

using namespace std;

class FindSameLengthWords

{

private:

char letter;

string in, s;

public:

void toStart(ifstream &inFile);

void OpenFileIn(ifstream &inFile, string nameIn);

int FileSize(ifstream& inFile, int fileSize);

void OpenFileOut(ofstream &outFile, string nameOut);

int CountWords(ifstream &inFile);

int CountLines(ifstream &inFile);

string ShowString(ifstream& inFile, string line, int numPages);

int StrLWord(ifstream &inFile, string line, int numPages, int numLetters);

};

/// FindSameLengthWords.сpp

#include "FindSameLengthWords.h"

void FindSameLengthWords :: toStart(ifstream &inFile)

{

inFile.clear();

inFile.seekg(0);

}

void FindSameLengthWords :: OpenFileIn(ifstream &inFile, string nameIn, int openError)

{

inFile.open(nameIn);

if (!inFile.is\_open())

{

//cout << "File can't be opened\n";

throw 1;

}

}

int FindSameLengthWords :: FileSize(ifstream& inFile, int fileSize)

{

inFile.seekg(0, ios::end);

fileSize = inFile.tellg();

toStart(inFile);

return fileSize;

}

void FindSameLengthWords :: OpenFileOut(ofstream &outFile, string nameOut)

{

outFile.open(nameOut);

}

int FindSameLengthWords :: CountWords(ifstream &inFile)

{

string item;

int count = 0;

#pragma omp parallel while

while (!inFile.eof())

{

inFile >> item;

count++;

}

toStart(inFile);

return count;

}

int FindSameLengthWords :: CountLines(ifstream &inFile)

{

char c;

int line\_count = 0;

#pragma omp parallel while

while (!inFile.eof())

{

c = inFile.get();

if (c == '\n')

{

line\_count++;

}

}

toStart(inFile);

return line\_count;

}

string FindSameLengthWords :: ShowString(ifstream& inFile, string line, int numPages)

{

for (int i = 1; i <= numPages; i++)

getline(inFile, line);

if (line.empty())

{

toStart(inFile);

return "String is empty, try another one. ";

}

else

return line;

}

int FindSameLengthWords :: StrLWord(ifstream &inFile, string line, int numPages, int numLetters)

{

char letter;

for (int i = 1; i <= numPages; i++)

getline(inFile, line);

int size = line.length();

string word;

int same\_length = 0;

#pragma omp parallel for

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

{

if (!isspace(line[i]) && isalpha(line[i]) && !isdigit(line[i]))

{

letter = line.at(i);

word += letter;

}

if (isspace(line[i]) || line[i] == '\0')

{

if (word.length() == numLetters)

{

word = "";

same\_length++;

}

word = "";

}

}

return same\_length;

}

/// main.сpp

#include "FindSameLengthWords.h"

int main() {

int sameLength = 0, sizeValue = 0;

int numLetters, numPages, fileSize = 0, openError=0;

string line, nameIn, nameOut;

FindSameLengthWords fileToRead;

ifstream inFile;

ofstream outFile;

cout << "Input name of file to read \*.txt: \t";

cin >> nameIn;

cout << "Input name of file to write in \*.txt: \t";

cin >> nameOut;

try

{

fileToRead.OpenFileIn(inFile, nameIn, openError);

fileToRead.OpenFileOut(outFile, nameOut);

cout << "File size: " << fileToRead.FileSize(inFile, fileSize) << " bt " << endl;

sizeValue = fileToRead.FileSize(inFile, fileSize);

if (sizeValue > 10000)

cout << "File is way too big, you need to wait a little bit \n";

//cout << fileToRead.CountWords(inFile) << " words in file" << endl;

cout << fileToRead.CountLines(inFile) << " strings in file" << endl;

cout << "S= "; cin >> numPages;

cout << "N= "; cin >> numLetters;

cout << fileToRead.ShowString(inFile, line, numPages) << endl;

cout << fileToRead.StrLWord(inFile, line, numPages, numLetters) <<

" words from string " << numPages << " contains " << numLetters << " letter" << endl;

outFile << fileToRead.StrLWord(inFile, line, numPages, numLetters);

}

catch (int openError) {

cout << "Cannot open file" << endl;

}

catch (string strl) {

cout << "Cannot process file"<< endl;

}

finally:

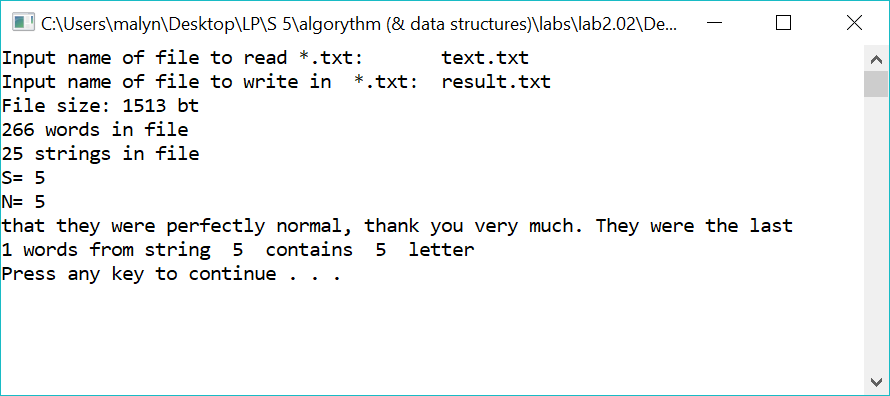
inFile.close();

outFile.close();

system("pause");

return 0;

}

 а.

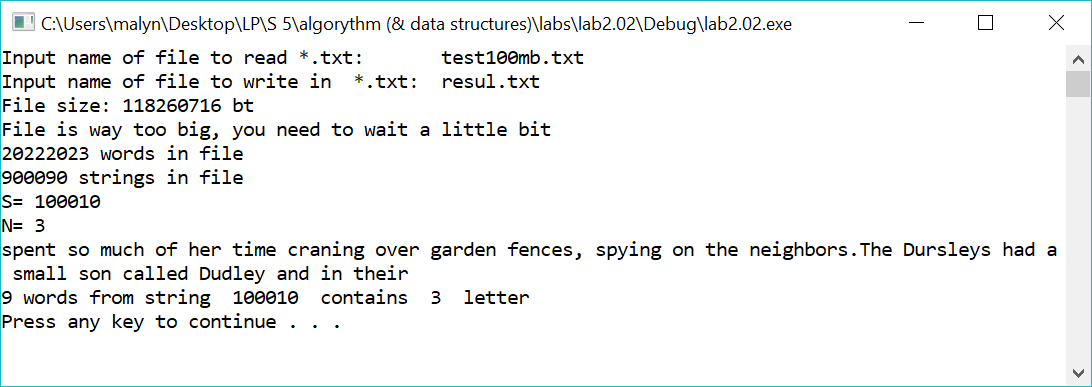
б.

Рис.2 Контрольні приклади

а) робота програми з відносно невеликим вхідним файлом

б) робота та відмінності роботи з великими файлами

**Висновок**

В результаті виконання цієї лабораторної роботи, я навчилася реалізовувати операції з файлами, зчитування та збереження інформації у них. Розробила програму, що визначає кількість слів у програмі, рядків, та визначає скільки слів однієї довжини є у певному рядку.