**Лабораторна робота 9**

**Тема: «Обробка текстової інформації»**

**9.3. Обробка текстової інформації, що міститься у файлах**

Завдання:

1. За допомогою текстового редактора створити файл, що містить текст. Довжина рядка тексту не повинна перевищувати 80 символів. Це вхідний файл.

2. Програма повинна

1) інформацію вхідного файлу записати у вихідний файл;

2) реалізувати дії, вказані в індивідуальному завданні і занести відповідну інформацію у вихідний файл.

3. Ім'я вхідного файлу задає користувач в командному рядку або під час виконання програми.

4. Ім'я вихідного файлу: перші символи - не більше трьох символів з імені вхідного файлу, а решти символів \_out.

5. Вихідний файл повинен мати розширення .dat.

6. Програма повинна забезпечувати по запиту користувача:

1) виведення тексту вхідного файлу на екран дисплея;

2) виведення тексту вихідного файлу на екран дисплея.

7. Закінчені послідовності дій оформити у вигляді функцій. Всі необхідні дані для функцій повинні передаватися їм як параметри. Використання глобальних змінних у функціях не допускається.

8. Для вводу і виводу інформації (і для консолі, і для текстових файлів) застосувати засоби форматованого вводу/виводу.

9. Визначення функцій і головну функцію розмістити в двох окремих файлах.

**20. Вивести рядки вхідного файлу додаючи, кількість слів в рядку, що закінчуються на голосну букву.**

Хід розв’язку

Щоб працювати з файлами потрібно було підключити бібліотеку fstream

Коли користувач введе ім’я файлу, потрібно перевірити чи є такий файл. Якщо такого файлу немає виведеться повідомлення про це.  
Після того як він введе правильну назву існуючого файлу, згенерується нова назва для іншого файлу, який буде створено пізніше. В функції для створення та зчитування передаються тільки назви файлів, хоча спочатку був варіант передавати посилання на об’єкти.

Головна функція перетворення строки має багато строкових функцій всередині. Для початку треба задати голосні літери, я записав їх до масиву (латинські та кирилицю) також треба розділяти слова, які входять до строки, я це зробив за допомогою функції strtok та використав ,.-?()!+\*<> такі символи для поділення на слова, кожне слово має кінець (\0) тому знаходимо порядок цього символу кінця і попередній символ це буде наша остання літера в слові, перевіряємо чи входить вона до нашого масиву з голосними літерами, якщо так то плюсуємо змінну count. Потім за допомогою sprintf і strcat з'єднуємо строки і записуємо нову строку до нового файлу

Код програми

//

// main.cpp

// prog-9.3

//

// Created by Дима Филипенко on 19.05.2020.

// Copyright © 2020 Дима Филипенко. All rights reserved.

//

#include <iostream>

#include <fstream>

**using** **namespace** std;

**int** openFile(**char** \*fileName){

fstream file;

file.open(fileName); // создаём объект класса ofstream для записи и связываем его с файлом

**if** (!file.is\_open()){

cout << "Файл не может быть открыт!\n";

**return** 1;

}

**return** 0;

}

**int** createNewName(**char** \*fileName,**char** \*NewfileName){

strncpy(NewfileName, fileName, 3);

NewfileName[3] = '\0';

strcat(NewfileName, "\_out.data");

**return** 0;

}

**int** openOutputFile(ofstream &file,**char** \*fileName){

**if** (!file.is\_open()){

cout << "Файл не может быть открыт!\n";

**return** 1;

}

**return** 0;

}

**char**\* getLine(fstream &file,**char** \*buffer){

file.getline(buffer,80);

**return** buffer;

}

**int** modifyLine(**char** \*str){

**char** oldstr[256];

strcpy (oldstr,str);

**char** vowels[]="EYUIOAУЕЇІАОЯИЮЄУЕЫОЯЮЭаоуеїієяиюыиэeyuioa";

**char** \* pch;

**char** \*lastletter;

**int** count=0;

**char** output[256];

pch = strtok (str," ,.-?()!+\*<>");

**while** (pch != **NULL**)

{

lastletter = strchr( pch, '\0' );

**if**(strstr (vowels,lastletter-1))

count++;

pch = strtok (**NULL**, " ,.-");

}

sprintf(output, "|| %d ", count);

strcat(oldstr,strcat(output," слів які закінчуються на голосну \n"));

strcpy (str,oldstr);

**return** count;

}

**int** getFileLines(**char** \*name){

**char** buffer[256];

fstream file(name);

**while** (file.getline(buffer,256)) {

cout<<buffer<<endl;

}

file.close();

**return** 0;

}

**int** func1(**char**\*name,**char**\*NewfileName,**char** \*buffer){

fstream fileIn(name);

fstream fileOut(NewfileName,ios\_base::out);

fileIn.ignore();

**while** (!fileIn.eof()) {

getLine(fileIn, buffer);

modifyLine(buffer);

fileOut<<buffer;

}

fileIn.close();

fileOut.close();

**return** 0;

}

**int** main(**int** argc, **const** **char** \* argv[]) {

//setlocale (LC\_ALL, "");

**char** fileName[80];

**char** NewfileName[80];

**char** buffer[256];

**int** k;

**do**{

cout<<"Введите имя файла:"<<endl;

cin>>fileName;

}**while**(openFile(fileName)==1);

createNewName(fileName,NewfileName);

**do**{

cout<<"1.Выполнить функцию"<<endl;

cout<<"2.Показать содержимое входного файла"<<endl;

cout<<"3.Показать содержимое выходного файла"<<endl;

cin>>k;

**if**(k==1)

func1(fileName,NewfileName,buffer);

**if**(k==2)

getFileLines(fileName);

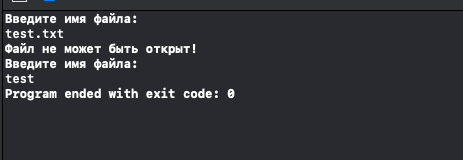
**if**(k==3)

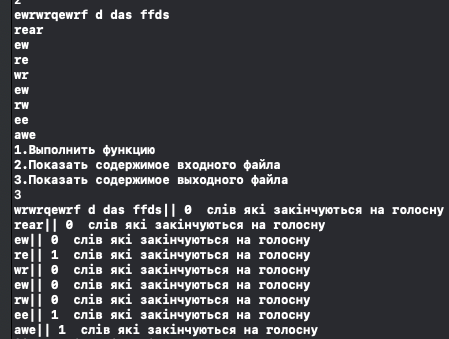
getFileLines(NewfileName);

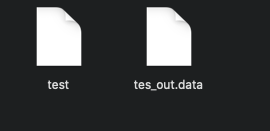
}**while**(k!=0);

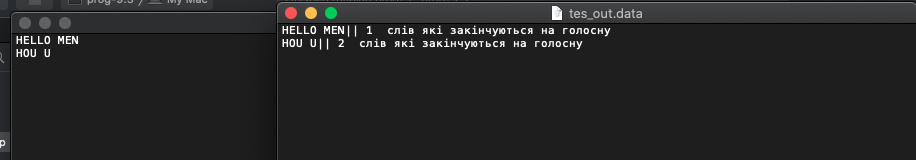
}

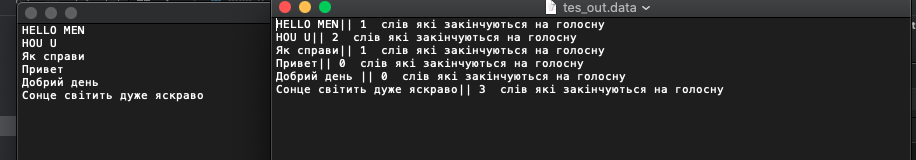
Тестові приклади

****

****

****

****

****

Керівництво користувача

На старті програми ви побачите що треба ввести назву файлу, який ви мали створити і записати туди декілька строк. Коли все успішно ви бачите меню 1 2 3

1 – це виконати функцію перетворення строк і запис їх в новий файл

2 – для того щоб подивитися що записано в файлі що створили ви

3 - для того щоб подивитися що записано в новому файлі, який створився після виконання пунку 1