ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ № 9

за курсом «Програмування»

студентки групи ПА-22-1

***Пономаренко Марини***

кафедра комп’ютерних технологій, ДНУ

2022/2023 н. р.

**Варіант 14**

**1.Постановка задачі**

**TASK 1**

Виконати задані дії із введенним текстом.

****

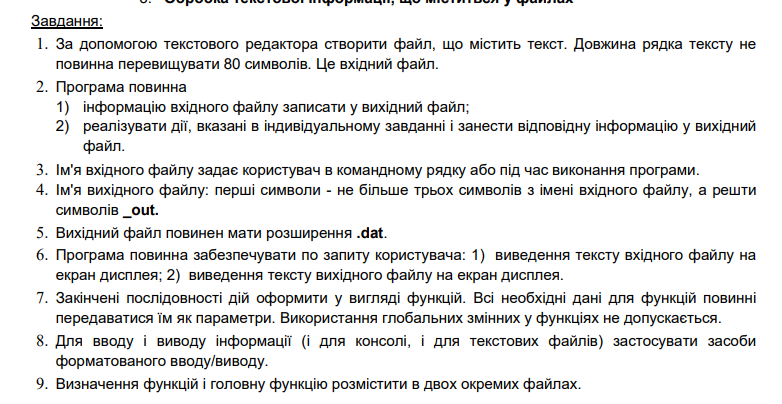
**TASK 2**

**Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание**

****

**TASK 3**

****

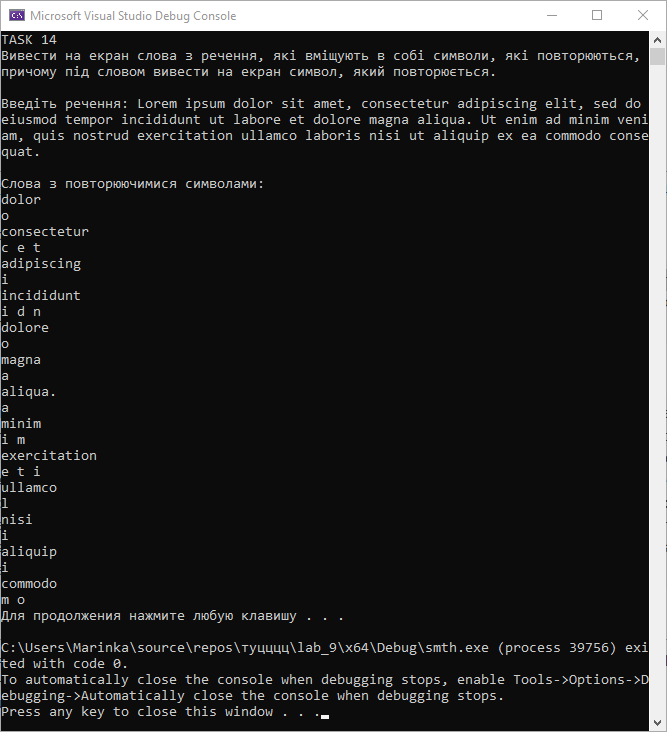
****

**2. Опис розв’язку**

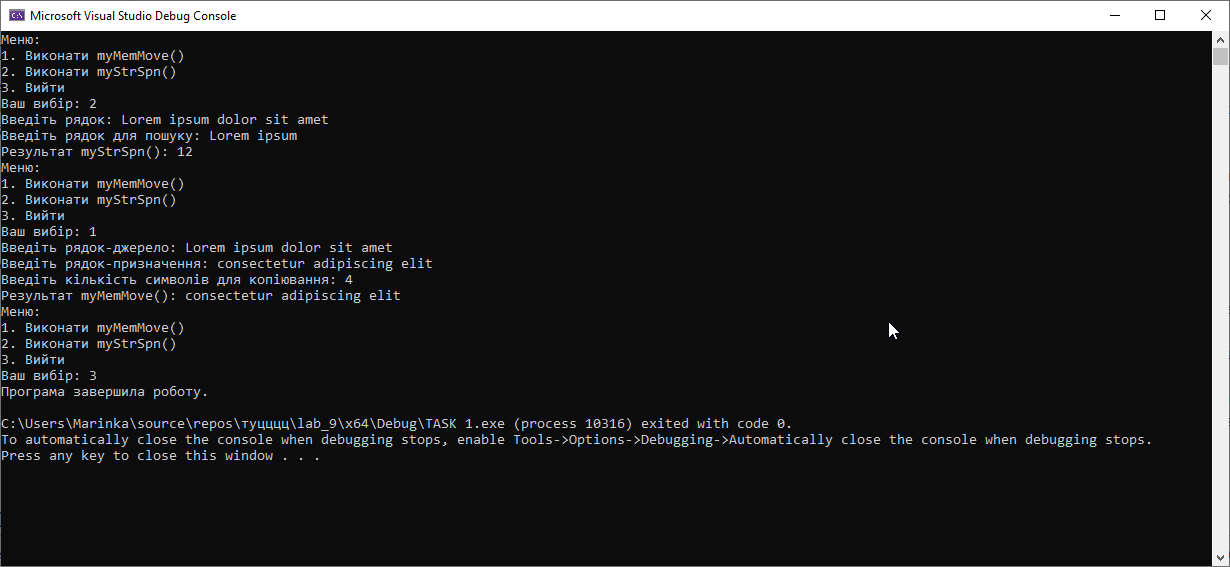
У цій лабораторній роботі, я працювала з IDE Visual Studio і мовою С++, застосувала навички, які здобула на лекціях з програмування.Ознайомлення із завданням, складання алгоритму, написання коду, перевірка на помилки.

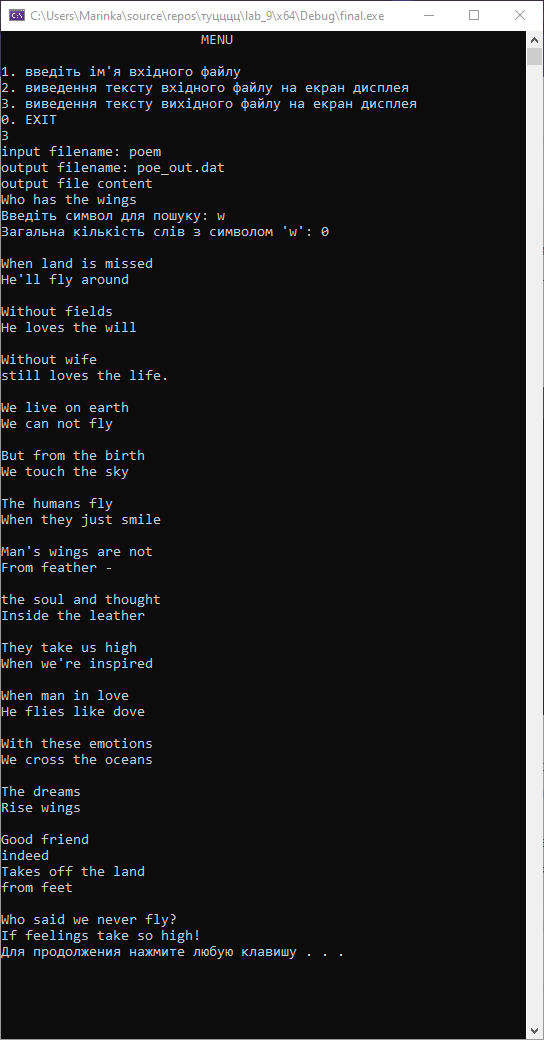
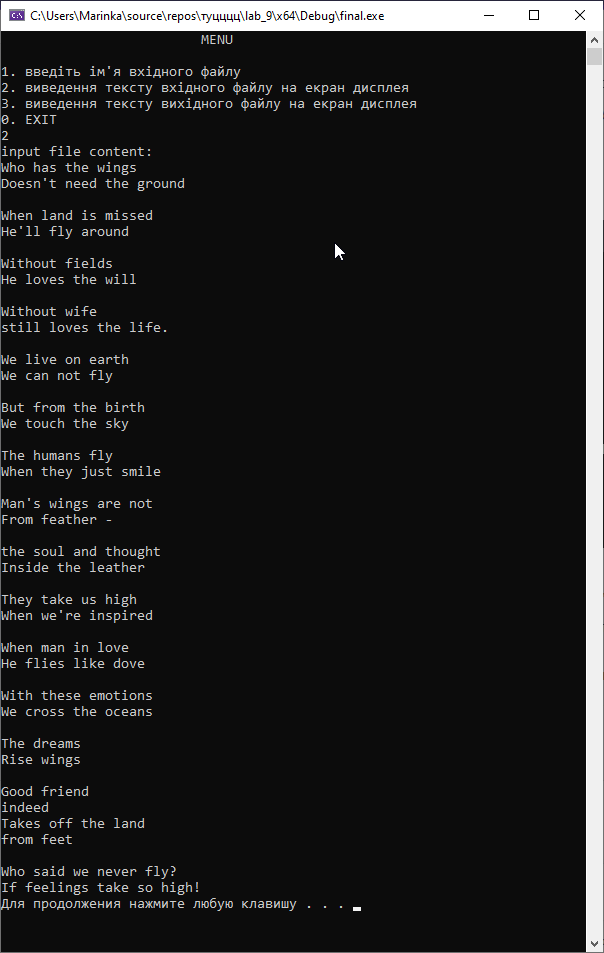
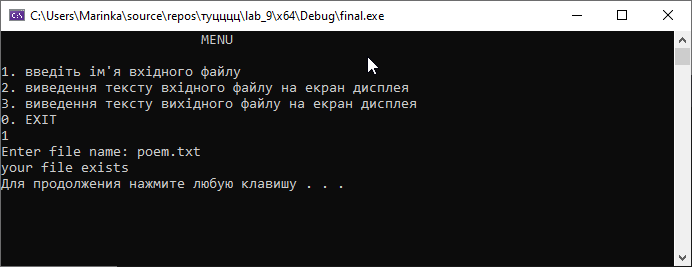
**3. Вихідний текст програми розв'язку задачі**

**TASK 1**

****

**TASK 2**

****

**TASK 3**

**4. Опис інтерфейсу програми**

**TASK 1**

#include <iostream>

#include <windows.h>

using namespace std;

char\* add(char a[], char x, int l)

{

char\* t = new char[l + 2];

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

t[i] = a[i];

t[l] = x;

t[l + 1] = '\0';

return t;

}

char\*\* add(char\*\* a, char\* x, int l)

{

char\*\* t = new char\* [l + 1];

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

t[i] = a[i];

t[l] = x;

return t;

}

int\* add(int a[], int x, int l)

{

int\* t{ new int[l + 1] {} };

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

t[i] = a[i];

t[l] = x;

return t;

}

char\* substr(char\* word, int begin, int lenght)

{

char\* res = new char[lenght + 1];

int l = 0;

for (int i = begin; i < begin + lenght; i++)

res[i - begin] = word[i];

res[lenght] = '\0';

return res;

}

bool ifAny(char ch, char\* symbols, int l)

{

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

{

if (ch == symbols[i]) return true;

}

return false;

}

bool ifAny(char\* s, char\*\* words, int l)

{

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

{

if (strcmp(s, words[i]) == 0) return true;

}

return false;

}

void fill(char\*\* line, int\* l)

{

char ch;

cin;

do

{

ch = cin.get();

\*line = add((\*line), ch, (\*l)++);

} while (ch != '\n');

}

void separate(char\*\* words, char\* line, int l, int\*\* lw, int\* cw)

{

int j = 0, k = 0;

bool space = false;

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

{

if (line[i] == ' ' || line[i] == '\n')

{

if (!space)

{

(\*lw) = add((\*lw), k, j++);

k = 0;

(\*cw)++;

space = true;

}

}

else

{

space = false;

words[j] = add(words[j], line[i], k++);

}

}

}

void repetitives() {

cout << "TASK 14" << endl;

cout << "Вивести на екран слова з речення, які вміщують в собі символи, які повторюються, причому під словом вивести на екран символ, який повторюється." << endl << endl;

char\* sentence = new char;

char \*\*words = new char\*;

int l = 0, \* lw = new int, cw = 0;

cout << "Введіть речення: ";

fill(&sentence, &l);

// Розділення речення на слова

separate(words, sentence, l, &lw, &cw);

cout << "\nСлова з повторюючимися символами: " << endl;

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

{

int l1 = 0, l2 = 0;

char\* reserved{ new char[l1] {} };

char\* repeat{ new char[l2] {} };

for (int j = 0; j < lw[i]; j++)

{

if (!ifAny(words[i][j], reserved, l1)) reserved = add(reserved, words[i][j], l1++);

else if (!ifAny(words[i][j], repeat, l2)) repeat = add(repeat, words[i][j], l2++);

}

if (l2 > 0)

{

cout << words[i] << endl;

for (int j = 0; j < l2; j++)

cout << repeat[j] << " ";

cout << endl;

}

}

}

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

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

repetitives();

system("pause");

}

**TASK 2**

#include <iostream>

#include <Windows.h>

using namespace std;

// Функція, яка імітує роботу memmove()

void myMemMove(void\* dest, const void\* src, size\_t count) {

char\* d = static\_cast<char\*>(dest);

const char\* s = static\_cast<const char\*>(src);

if (d < s) {

for (size\_t i = 0; i < count; i++) {

d[i] = s[i];

}

}

}

// Функція, яка імітує роботу strspn()

size\_t myStrSpn(const char\* str1, const char\* str2) {

const char\* s1 = str1;

size\_t count = 0;

while (\*s1) {

const char\* s2 = str2;

bool found = false;

while (\*s2) {

if (\*s1 == \*s2) {

found = true;

break;

}

s2++;

}

if (!found) {

break;

}

count++;

s1++;

}

return count;

}

int main() {

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

char choice;

char source[100];

char dest[100];

char searchSet[100];

size\_t result;

do {

cout << "Меню:" << endl;

cout << "1. Виконати myMemMove()" << endl;

cout << "2. Виконати myStrSpn()" << endl;

cout << "3. Вийти" << endl;

cout << "Ваш вибір: ";

cin >> choice;

switch (choice) {

case '1':

cout << "Введіть рядок-джерело: ";

cin.ignore();

cin.getline(source, sizeof(source));

cout << "Введіть рядок-призначення: ";

cin.getline(dest, sizeof(dest));

int count;

cout << "Введіть кількість символів для копіювання: ";

cin >> count;

myMemMove(dest, source, count);

cout << "Результат myMemMove(): " << dest << endl;

break;

case '2':

cout << "Введіть рядок: ";

cin.ignore();

cin.getline(source, sizeof(source));

cout << "Введіть рядок для пошуку: ";

cin.getline(searchSet, sizeof(searchSet));

result = myStrSpn(source, searchSet); // Оновлено result

cout << "Результат myStrSpn(): " << result << endl;

break;

case '3':

cout << "Програма завершила роботу." << endl;

break;

default:

cout << "Некоректний вибір. Спробуйте ще раз." << endl;

break;

}

} while (choice != '3');

return 0;

}

**TASK 3**

#define NOMINMAX

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <cstring>

#include <conio.h>

#include <Windows.h>

#include <fstream>

using namespace std;

#include "Functions.h"

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

char FilePath[101] = "";

int choice = 4;

while (choice != 0) {

cout << "\t\t\t MENU \n";

cout << "\n";

cout << "1. введіть ім'я вхідного файлу \n";

cout << "2. виведення тексту вхідного файлу на екран дисплея\n";

cout << "3. виведення тексту вихідного файлу на екран дисплея\n";

cout << "0. EXIT \n";

cin >> choice;

if (!(choice == 0 or choice == 1 or choice == 2 or choice == 3)) {

cout << "error choose the correct number\n";

system("pause");

}

else {

if (choice == 0) {

cout << "\n";

system("pause");

}

if (choice == 1) {

cout << "Enter file name: ";

cin.ignore(numeric\_limits<streamsize>::max(), '\n');//чтобы гетлайн норм работал

cin.getline(FilePath, 100);

ifstream inputFile(FilePath);

while (inputFile.fail()) {

inputFile.close();

cout << "Error your file does not exist!!!\n";

cout << "Enter file name: ";

//

cin.ignore(numeric\_limits<streamsize>::max(), '\n');//чтобы гетлайн норм работал

cin.getline(FilePath, 100);

inputFile.open(FilePath);

}

inputFile.close();

cout << "your file exists\n";

system("pause");

}

if (choice == 2) {

if (FilePath == nullptr or strlen(FilePath) == 0) {

cout << "you did not input filename \n";

}

else {

ifstream inputFile(FilePath);

cout << "input file content:\n";

char str[100];

while (!inputFile.eof()) {

inputFile.getline(str, 100);

cout << str << endl;

}

cin.ignore(numeric\_limits<streamsize>::max(), '\n');//чтобы гетлайн норм работал

inputFile.close();

}

system("pause");

}

if (choice == 3) {

if (FilePath == nullptr or strlen(FilePath) == 0) {

cout << "you did not input filename \n";

}

else {

char FileName[101]; char outputFilename[] = "^^^\_out.dat";

cout << "input filename: " << getFileName(FilePath, FileName) << endl;

int index = 0;

for (int i = 0; i < 3 && i < strlen(FileName); i++) {

outputFilename[i] = FileName[i];

index++;

}

size\_t outputFilenamelengh = strlen(outputFilename) - (3 - static\_cast<unsigned long long>(index));

for (int i = 0; index < outputFilenamelengh; i++) {

outputFilename[index] = outputFilename[i + 3];

index++;

}

outputFilename[index] = '\0';

cout << "output filename: " << outputFilename << endl;

ofstream outFile; // поток для записи

outFile.open(outputFilename); // окрываем файл для записи

ifstream inputFile(FilePath);

cout << "output file content \n";

char str[100];

int stringNumber = 1;

while (!inputFile.eof()) {

inputFile.getline(str, 100);

if (stringNumber == 2) {

char searchChar;

cout << "Введіть символ для пошуку: ";

cin >> searchChar;

findWords(FilePath, searchChar);

//cout << str << endl;

outFile << str << '\n';

}

else {

cout << str << endl;

outFile << str << '\n';

}

stringNumber++;

}

//

cin.ignore(numeric\_limits<streamsize>::max(), '\n');//чтобы гетлайн норм работал

inputFile.close();

outFile.close();

}

system("pause");

}

}

system("cls");

}

return 0;

}