Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет комп’ютерних систем та автоматики

Кафедра КСУ

Лабораторна робота №2

з дисципліни: «Компютерні технології та програмування»

Тема: «Обробка символьної інформації»

Виконав

ст. гр. 2АКІТ-20б

О. Мананов

Перевірив

к.т.н., доц.кафедри КСУ

Севастьянов В. М.

м.Вінниця, ВНТУ 2021

**Лабораторна робота №2**

**Тема: «Обробка символьної інформації»**

**Завдання :**

1. Створити програму на мові С згідно варіанту використавши середовище

програмування Dev-C++ 4.0:

* за допомогою текстового редактора створити файл, що містить текст, довжина якого не перевищує 1000 символів (довжина рядка тексту не повинна перевищувати 70 символів);
* ім'я файлу повинне мати розширення dat.

1. Відкомпілювати та відлагодити програму.
2. Розробити набір тестів і перевірити роботу програми на них.
3. Відповісти на контрольні запитання.
4. Зробити висновки.

**Варіант 10**

Написати програму, яка виводить на екран символ, який зустрічається у файлі максимальне число разів.

**Хід роботи :**

**1. Ретельно вивчаю поставлене перед мною завдання.**

**2. Відкриваю середовище та пишу власне сам код програми за допомогою середовища Dev-C++.**

**3. Додаю до програми необхідні бібліотеки, додаю середовище імен using namespace** std**;. А також ініціалізую функцію** mostUsed()**;**

#include <string>

#include <vector>

#include <iostream>

#include <fstream>

#include <Windows.h>

using namespace std;

char mostUsed(string str);

**4. Описую головну функцію** int main**() з можливістю введення кирилці кодування Windows-1251.**

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

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

**5. Тут же ініціалізую стрічковий** string str**, для зчитування в нього тексту з файла, символний** path[BUFSIZ], **для введення місцеположення в системі самого файла і його подальшого відкриття (має BUFSIZ, де макс. кількість введених символів = розміру буфера системи)**

string str;

char path[BUFSIZ];

cout << "Введіть шлях до файлу (Приклад : D:\\\\КТП\\\\l2.dat) :\n";

cin >> path;

cout << "Текст в файлі " << path << " :\n\n";

**6. Функція ifstream призначена для роботи з файлами, тому її необхідно використати для відкриття текстового файла у форматі .dat**

ifstream read\_n\_check(path, std::ios::app);

str.assign( (std::istreambuf\_iterator<char>(read\_n\_check) ), (std::istreambuf\_iterator<char>() ) );

**7. Нижче написані стандартні команди cout, для виведення на екран пристрою данних файла, а також застосування функції** mostUsed();**, де визначається найбільш поширений в файлі символ.**

cout << str << std::endl;

cout << "\nСимвол, що зустрічається максмальну кількість разів : " << mostUsed(str);

read\_n\_check.close();

return 0;

}

**8. Функція** mostUsed();**, за допомогою векторів (аналогів динамічних масивів) порівнює кількість символів та рахує найбільш поширеніший і повертає** return letters[maxIndex]; **змінну типу char.**

char mostUsed(string str)

{

vector<char> letters;

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

{

if (i == 0)

{

letters.push\_back(str[i]);

continue;

}

int count = 0;

for (int j = 0; j < letters.size(); j++)

{

if (str[i] == letters[j])

break;

else

{

for (int k = 0; k < letters.size(); k++)

{

if (str[i] != letters[k])

count++;

}

}

if (count == letters.size())

letters.push\_back(str[i]);

}

}

vector<int> occur(letters.size());

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

{

for (int j = 0; j < letters.size(); j++)

{

if (str[i] == letters[j])

occur[j]++;

}

}

int maxAmount = -1, maxIndex = -1;

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

{

if (occur[i] > maxAmount)

{

maxAmount = occur[i];

maxIndex = i;

}

}

return letters[maxIndex];

}

**11. Шукаю та виправляю помилки, оптимізую код.**

**Висновок :** виконуючи дану лабораторно роботу, я навчився користуватись бібліотеками vector та fstream, читати\записувати інформацію у файл та сортувати символьні елементи завдяки векторним масивам.