НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

КАФЕДРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



**ЗВІТ**

**Про виконання лабораторної роботи № 10**

**з дисципліни «Операційні системи»**

**Лектор:**

Ст. викладач Грицай О.Д.

**Виконав:**

студ. групи ПЗ-23

Михалевич П.-І.В.

**Прийняла:**

Ст. викладач Грицай О.Д.

«\_\_\_» \_\_\_\_\_\_ 2021 р.

∑ = \_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_

Львів – 2021

**Тема.** Управління файловою системою.

**Мета.** Ознайомитися з файловими системами операційних систем Windows та LINUX.

**Теоретичні відомості**

Файлова система – це система правил, згідно яких операційна система забезпечує зберігання даних на диску. Ці правила визначають формат зберігання даних, їх розміщення на диску, адресацію і ідентифікацію. Кожен комп'ютерний файл зберігається на цифровому носії із певною місткістю. Накопичувач можна уявити як лінійний простір для читання і запису цифрової інформації. Кожен байт даних на ньому має своє зміщення від початку сховища, відомий як адреса, яка використовується для посилання на цей файл. Накопичувачі для посилання на байт інформації можуть використовувати адресацію у вигляді пари - сектор і зміщення. Загалом, файлова система це структуроване представлення даних та набір метаданих, що описують ці дані. Вона записується до сховища під час операції форматування. Зазвичай ця структура оперує блоками, а не секторами. Блоки ФС – це групи секторів, за допомогою яких оптимізується адресація сховища. Файли, як правило, зберігаються на початку блоку та займають цілі блоки. Крім даних користувача, файлова система також містить власні параметри (наприклад, розмір блоку), файлові дескриптори (такі як його розмір, розташування, фрагменти і т.д.), імена та ієрархію каталогів. Вона також може зберігати інформацію про безпеку, розширені атрибути та інші параметри. Щоб відповідати різноманітним вимогам користувачів, таким як продуктивність сховища, стабільність і надійність, розробляється безліч типів (або форматів) файлових систем, здатних більш ефективно виконувати певні функції.

**Хід роботи**

**Завдання**

1. Створити за допомогою API функцій файл для запису результатів виконання лабораторної роботи №3 (відповідно до свого варіанту).

2. Створити файл для запису результатів виконання лабораторної роботи №4 (відповідно до свого варіанту).

3. Реалізувати зміну прав доступу користувача до файлу.

4. Результати виконання роботи відобразити у звіті.

**Індивідуальне завдання**

**Windows:**

**Код програми:**

#include <iostream>

#include <vector>

#include <ctime>

#include <windows.h>

#include <sstream>

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

{

HANDLE hConsole = GetStdHandle(STD\_OUTPUT\_HANDLE);

int N;

int M;

std::stringstream str;

srand(time(NULL));

try {

if (argc >= 2) {

std::istringstream stdStream\_1(argv[1]);

std::istringstream stdStream\_2(argv[2]);

if (!(stdStream\_1 >> N))

{

throw 1;

}

if (!(stdStream\_2 >> M))

{

throw 1;

}

}

}

catch(int i){

if (i == 1) {

std::cout << std::endl <<"Invalid input";

return 1;

}

}

/\*

do {

std::cout << "Enter the N :";

std::cin >> N;

if (N <= 1000) std::cout << "Error! Enter the right number." << std::endl;

else break;

} while (1);

do {

std::cout << std::endl << "Enter the M :";

std::cin >> M;

if (M <= 10000) std::cout << "Error! Enter the right number." << std::endl;

else break;

} while (1);

\*/

std::vector<std::vector<short>> matr(N, std::vector<short>(M,0));

for (int i = 0; i < N; i++) {

for (int j = 0; j < M; j++) {

matr[i][j] = rand() % 100;

}

}

std::vector<int> numberOfCount;

for (int i = 0; i < M; i++) {

int n = 0;

for (int j = 1; j < N; j++) {

if (matr[0][i] == matr[j][i]) {

++n;

}

}

numberOfCount.push\_back(n);

}

int k = 0;

for (int i = 0; i < M; i++, k++)

{

if (k % 7 == 6) {

SetConsoleTextAttribute(hConsole, 0x05);

}

else if (k % 7 == 5) {

SetConsoleTextAttribute(hConsole, 0x03);

}

else if (k % 7 == 4) {

SetConsoleTextAttribute(hConsole, 0x01);

}

else if (k % 7 == 3) {

SetConsoleTextAttribute(hConsole, 0x02);

}

else if (k % 7 == 2) {

SetConsoleTextAttribute(hConsole, 0x06);

}

else if (k % 7 == 1) {

SetConsoleTextAttribute(hConsole, 0x0C);

}

else if (k % 7 == 0) {

SetConsoleTextAttribute(hConsole, 0x04);

}

std::stringstream str\_1;

str << "The number of similar to first element in " << i << " column is "<< numberOfCount[i] << std::endl;

str\_1 << "The number of similar to first element in " << i << " column is "<< numberOfCount[i] << std::endl;

std::string stringline = str\_1.str();

std::cout << stringline;

}

HANDLE file = CreateFile(L"C:\\LPNU\\os\\labs\\lab3\\OS\_lab\_3\\File.txt", GENERIC\_WRITE, FILE\_SHARE\_WRITE, NULL, CREATE\_ALWAYS, FILE\_ATTRIBUTE\_NORMAL, NULL);

if (file == INVALID\_HANDLE\_VALUE)

{

std::cout << "Can't open the file!" << std::endl;

system("pause");

return 2;

}

std::string buf = str.str();

//HANDLE map = CreateFileMappingW(file, NULL, PAGE\_READWRITE, NULL, buf.size(), NULL);

//LPVOID pointer = MapViewOfFile(map, FILE\_MAP\_ALL\_ACCESS, 0, 0, buf.size());

//memcpy(pointer, buf.c\_str(), buf.size());

//UnmapViewOfFile(pointer);

//CloseHandle(map);

WriteFile(file, buf.c\_str(), buf.size(), NULL, NULL);

CloseHandle(file);

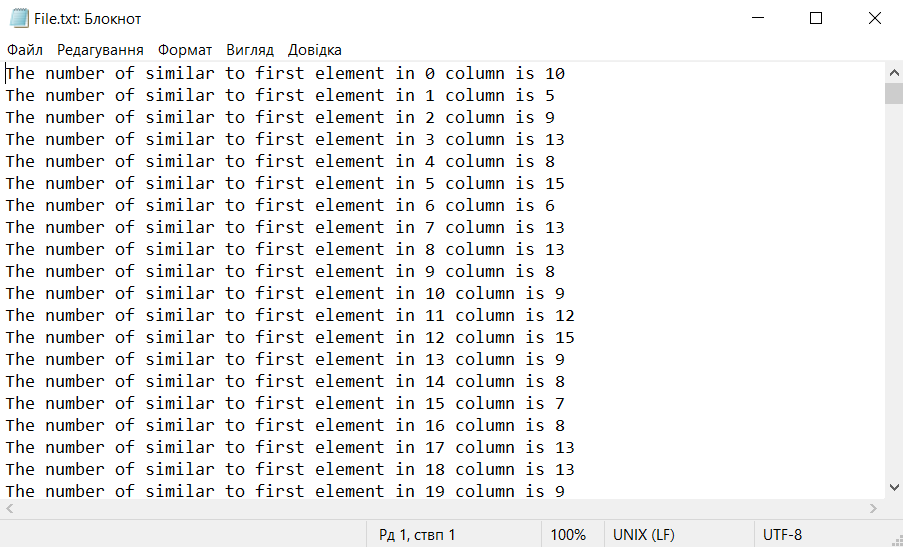
SetConsoleTextAttribute(hConsole, 0xFFFF);

system("pause");

return 0;

}

**Результат програми:**

****

**Linux:**

**Код програми:**

#include <iostream>

#include <vector>

#include <ctime>

#include <sstream>

#include <sys/shm.h>

#include <fcntl.h>

#include <sys/mman.h>

#include <sys/stat.h>

#include <sys/types.h>

#include <unistd.h>

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

{

int N;

int M;

std::stringstream str;

srand(time(NULL));

try {

if (argc >= 2) {

std::istringstream stdStream\_1(argv[1]);

std::istringstream stdStream\_2(argv[2]);

if (!(stdStream\_1 >> N))

{

throw 1;

}

if (!(stdStream\_2 >> M))

{

throw 1;

}

}

}

catch(int i){

if (i == 1) {

std::cout << std::endl <<"Invalid input";

return 1;

}

}

std::vector<std::vector<short>> matr(N, std::vector<short>(M,0));

for (int i = 0; i < N; i++) {

for (int j = 0; j < M; j++) {

matr[i][j] = rand() % 100;

}

}

std::vector<int> numberOfCount;

for (int i = 0; i < M; i++) {

int n = 0;

for (int j = 1; j < N; j++) {

if (matr[0][i] == matr[j][i]) {

++n;

}

}

numberOfCount.push\_back(n);

}

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

{

std::stringstream str\_1;

//str << "The number of similar to first element in " << i << " column is "<< numberOfCount[i] << std::endl;

str\_1 << "The number of similar to first element in " << i << " column is "<< numberOfCount[i] << "\n";

std::string stringline = str\_1.str();

std::cout << stringline;

const char\* buf = str\_1.str().c\_str();

const int size = str\_1.str().size();

int fd;

if((fd = open("//home//pavlo//source1\_lab\_4//file.txt", O\_CREAT | O\_RDWR | O\_APPEND, S\_IRUSR | S\_IWUSR)) < 0){

std::cout << "create() error";

}else{

write(fd,buf,size);

close(fd);

}

}

//int d = shm\_open("name", O\_CREAT | O\_RDWR, 0666);

//ftruncate(buf, size);

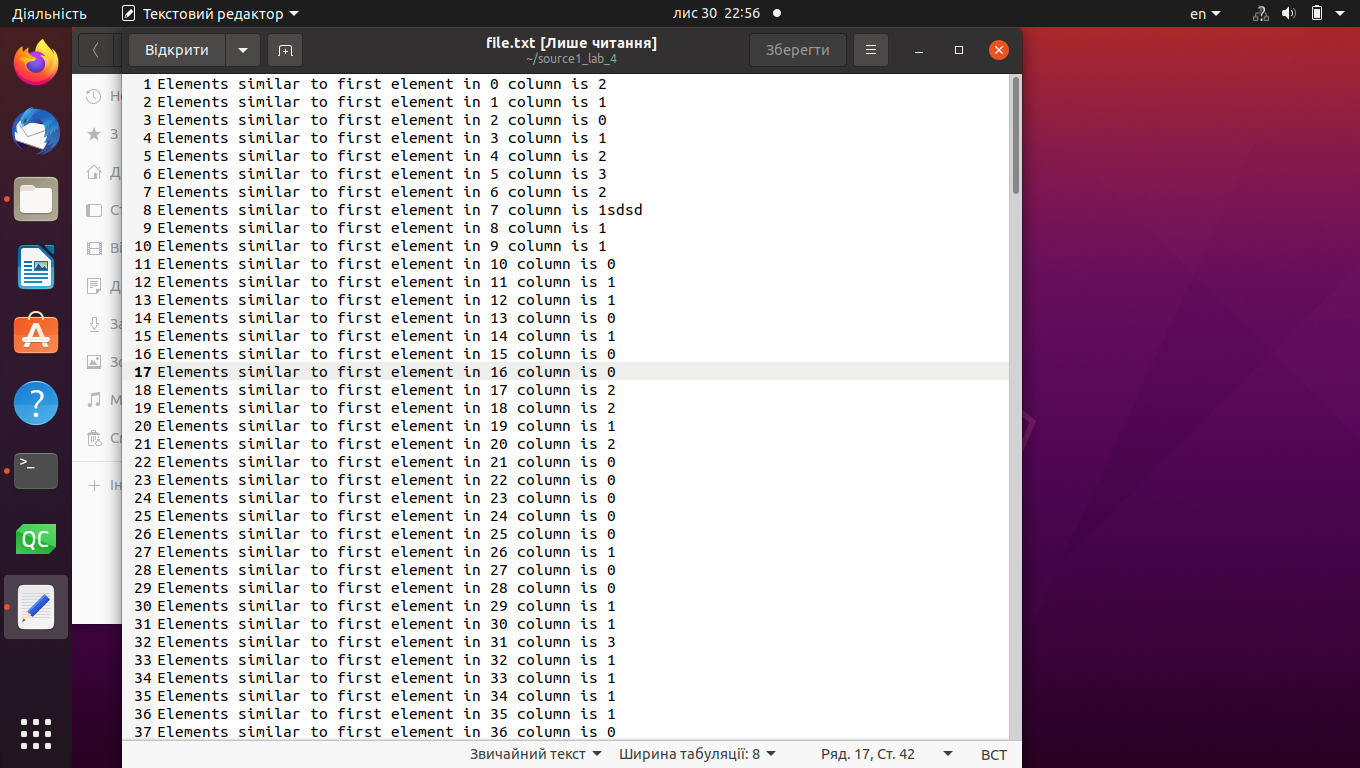
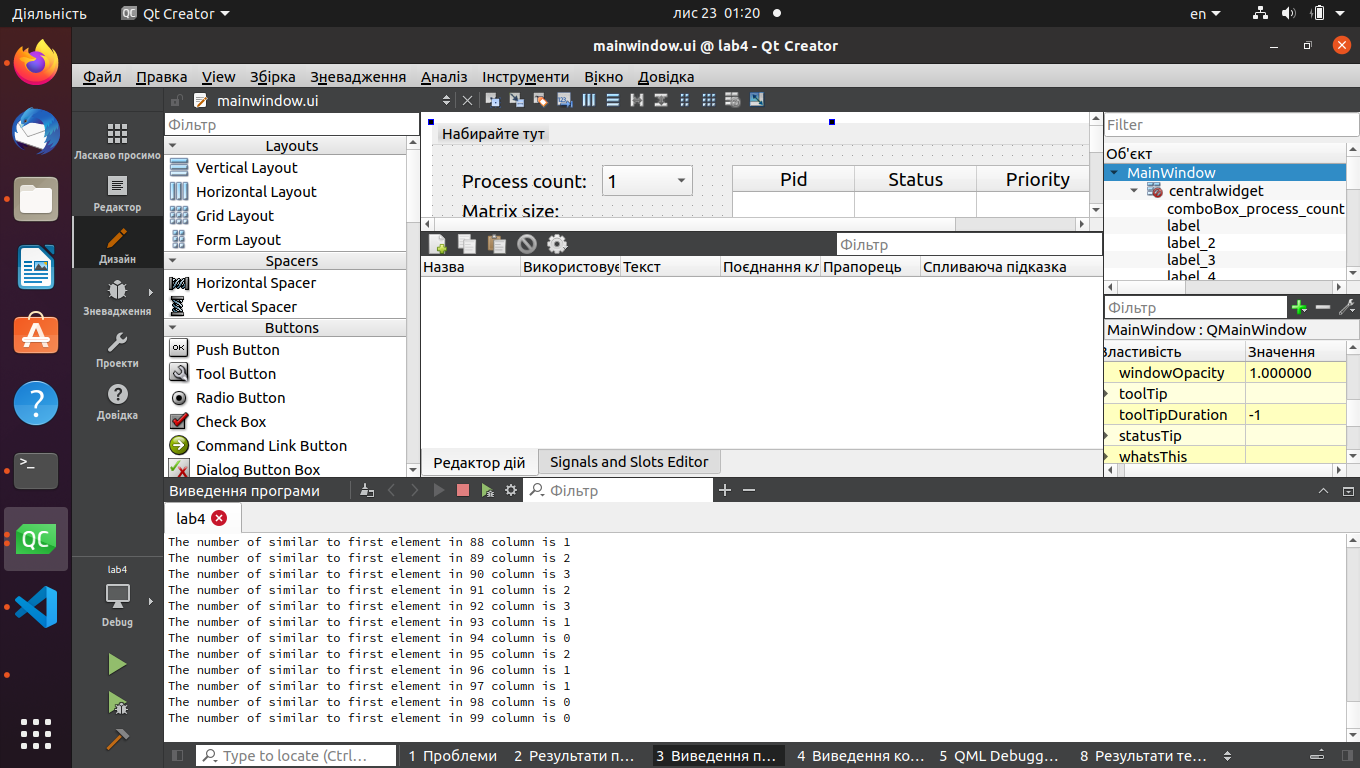
//char\* cPtr = (char\*)mmap(NULL, size, PROT\_READ | PROT\_WRITE, MAP\_SHARED, d, 0);

//sprintf(cPtr, "%s", buf);

return 0;

}

**Результат програми:**

****

##### Висновок

На даній лабораторній роботі, я ознайомився з файловими системами операційних систем Windows та LINUX. Реалізував програму, яка записує результат лаб роботи №3 і №4 у файл.