Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут»

Навчально-науковий комплекс «Інститут прикладного системного аналізу»

Кафедра системного проектування

Лабораторна робота № 3

з курсу «Операційні системи»

Виконала студентка

Групи ДА-02

Лесечко О. Р

Київ 2022

**Завдання:**

Написати власну версію програми ls для Unix. Ця версія повинна приймати як параметр одне або кілька імен каталогів і для кожного каталогу видавати список усіх файлів, що містяться в цьому каталозі, по одному рядку на файл. Для кожного файлу повинна бути виведена інформація про його ім'я, розмір, тип, власника, права доступу та час доступу. Кожне поле має форматуватися відповідно до його типу. Вкажіть у списку лише першу дискову адресу (або жодної).

**Код програми:**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include <chrono>

#include <iomanip>

#include <fstream>

#include <filesystem>

using namespace std::chrono\_literals;

// для того, щоб бачити дату останньої зміни файла

template <typename TP>

std::time\_t to\_time\_t(TP tp)

{

using namespace std::chrono;

auto sctp = time\_point\_cast<system\_clock::duration>(tp - TP::clock::now() + system\_clock::now());

return system\_clock::to\_time\_t(sctp);

}

// для того, щоб бачити який доступ є у файла

void demo\_perms(std::filesystem::perms p)

{

std::cout << ((p & std::filesystem::perms::owner\_read) != std::filesystem::perms::none ? "r" : "-")

<< ((p & std::filesystem::perms::owner\_write) != std::filesystem::perms::none ? "w" : "-")

<< ((p & std::filesystem::perms::owner\_exec) != std::filesystem::perms::none ? "x" : "-")

<< ((p & std::filesystem::perms::group\_read) != std::filesystem::perms::none ? "r" : "-")

<< ((p & std::filesystem::perms::group\_write) != std::filesystem::perms::none ? "w" : "-")

<< ((p & std::filesystem::perms::group\_exec) != std::filesystem::perms::none ? "x" : "-")

<< ((p & std::filesystem::perms::others\_read) != std::filesystem::perms::none ? "r" : "-")

<< ((p & std::filesystem::perms::others\_write) != std::filesystem::perms::none ? "w" : "-")

<< ((p & std::filesystem::perms::others\_exec) != std::filesystem::perms::none ? "x" : "-")

<< '\n';

}

// власна версія програми dir

void ls(std::filesystem::path path) {

try {

// рекурсивный обход каталога

for (const std::filesystem::path& path\_tmp : std::filesystem::directory\_iterator(path)) {

if(!std::filesystem::is\_directory(path\_tmp)){

// інформація про ім'я файла

std::cout << "name: " << path\_tmp.filename() << "\n";

// інформація про розмір файла

std::cout << "size: " << std::filesystem::file\_size(path\_tmp) << "\n";

// інформація про тип файла

std::cout << "extension: " << path\_tmp.extension() << "\n";

// інформація про права доступу файла

std::cout << "access: "; demo\_perms(std::filesystem::status(path\_tmp).permissions());

// інформація про час доступу

auto ftime = std::filesystem::last\_write\_time(path\_tmp);

std::time\_t cftime = to\_time\_t(ftime);

std::cout << "File write time is " << std::asctime(std::localtime(&cftime)) << "\n\n";

}

}

}

catch (std::filesystem::filesystem\_error& e) {

std::cerr << e.what() << std::endl;

}

}

int main() {

ls("/home/kenobi");

return 0;

}

**Результат роботи:**

Результат роботи це список усіх файлів, що містяться в каталозі. Для кожного файлу виведена інформація про його ім'я, розмір, тип, права доступу та час доступу.

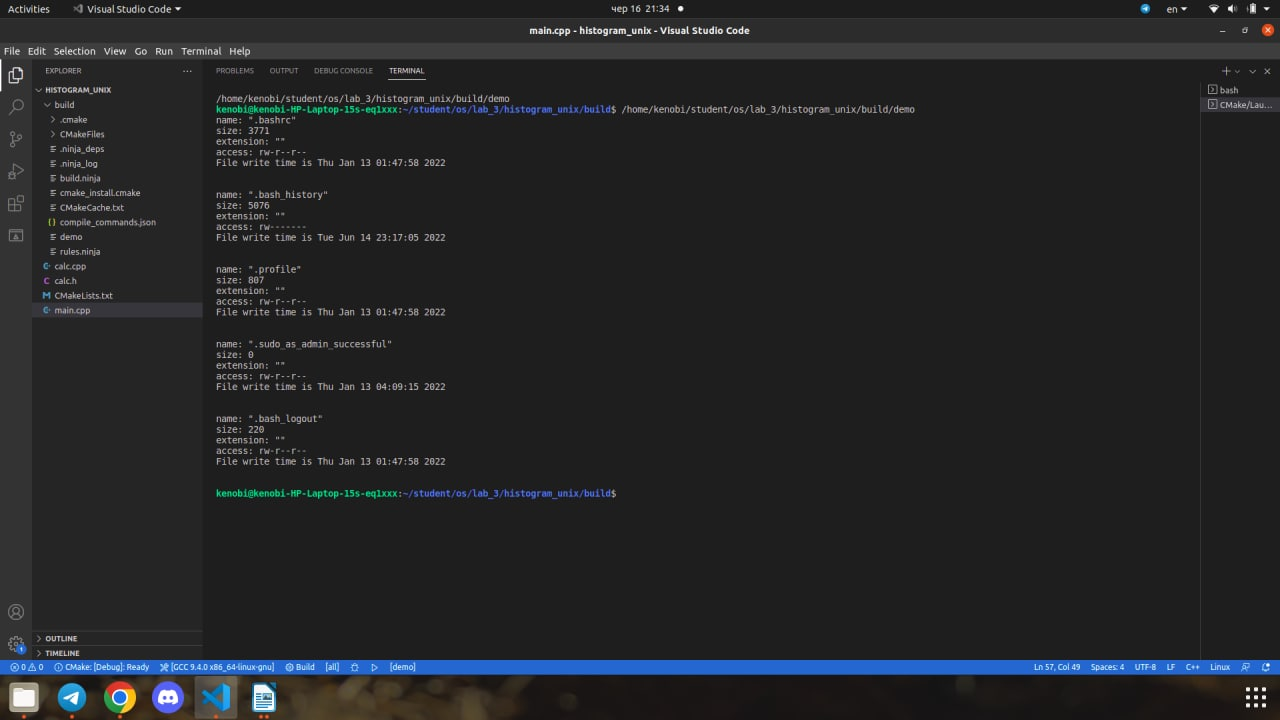


Рис. 2 Результат роботи програми

**Висновки:**

Написала власну версію програми ls для Unix. Ця версія повинна приймає як параметр одне або кілька імен каталогів і для кожного каталогу видає список усіх файлів, що містяться в цьому каталозі. Для кожного файлу повинна бути виведена інформація про його ім'я, розмір, тип, права доступу та час доступу.