

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное учреждение высшего образования
«Чувашский государственный университет И.Н. Ульянова»
Факультет информатики и вычислительной техники
Кафедра вычислительной техники

Системное программирование
Лабораторная работа 6
«Управление файла и каталогами»

Выполнил:

Студент группы ИВТ-41-20
Галкин Д.С.

Проверил:

Яковлев С.В.

Цель работы:

Напишите и выполните программу **TestFileDir** Имя1 Имя2, которая демонстрирует использование основных функций обработки файлов и каталогов и на основе меню позволяет пользователю создавать, удалять, копировать, переименовывать файлы и подкаталоги текущего каталога.

Ограничения:

1. По ключу `"/?"` в командной строке программа должна выдавать справку о своем назначении, использовании и авторе
2. Имена файлов или каталогов источника и приемника (если требуется) задаются в командной строке
3. Меню создавать в консольном приложении путем вывода строк - пунктов меню с нумерацией
4. Выбор пункта меню выполняется путем ввода его номера в интерактивном режиме
5. Программа должна запрашивать подтверждение переименования файла или каталога, а также сообщать об ошибках отсутствия файла (каталога) и невозможности выполнить требуемый функцию (пункт меню)
6. После сообщения о любых ошибках меню должно повторно выводиться на экран вместе с именами обрабатываемых файлов (каталогов)

Полный текст программы:

file_systems_utils.h

```
//  
// Created by Dmitry Galkin on 16.02.2024.  
//  
  
#ifndef FILE_SYSTEMS_UTILS_H  
#define FILE_SYSTEMS_UTILS_H  
#include <string>  
  
// Вывод справочника  
void printHelp();  
// Создание файла или каталога  
void create(const std::string& name);  
// Удаления файла или каталога  
void remove(const std::string& name);  
// Копирование файла или каталога  
void copy(const std::string& source, const std::string& destination);  
// Переименование файла или каталога  
void rename(const std::string& oldName, const std::string& newName);  
  
#endif //FILE_SYSTEMS_UTILS_H
```

```

//
// Created by Dmitry Galkin on 16.02.2024.
//

#include "file_systems_utils.h"
#include <iostream>
#include <string>
#include <filesystem> // Требуется для C++17 и выше
#include <fstream>

namespace fs = std::filesystem;
const fs::path PATH_FILES_LAB6 = "../Lab6/resources/";

void printHelp() {
    std::cout << "TestFileDir – программа для демонстрации работы с файлами и каталогами.\n";
    std::cout << "Использование: TestFileDir Имя1 Имя2\n";
    std::cout << "Команды:\n";
    std::cout << "1 – Создать файл/каталог\n";
    std::cout << "2 – Удалить файл/каталог\n";
    std::cout << "3 – Копировать файл/каталог\n";
    std::cout << "4 – Переименовать файл/каталог\n";
    std::cout << "5 – Выйти из программы\n";
    std::cout << "Для получения справки используйте ключ /?.\n";
}

void create(const std::string& name) {
    fs::path PATH_FILES_LAB6;
    // Проверяем, требуется ли создать каталог или файл
    if (fs::is_directory(PATH_FILES_LAB6 / name) || name.find('.') ==
std::string::npos) {
        // Пытаемся создать каталог
        try {
            if (fs::create_directory(PATH_FILES_LAB6 / name)) {
                std::cout << "Каталог создан: " << name << std::endl;
            } else {
                std::cout << "Не удалось создать каталог: " << name << std::endl;
            }
        } catch (const fs::filesystem_error& e) {
            std::cout << "Ошибка при создании каталога: " << e.what() << std::endl;
        }
    } else {
        // Пытаемся создать файл
        // Проверяем, существует ли каталог if (!fs::exists(PATH_FILES_LAB6)) {
        std::cout << "Каталог для создания файла не существует: " <<
PATH_FILES_LAB6 << std::endl;
        } else {
            std::ofstream file(PATH_FILES_LAB6 / name);
            if (file) {
                std::cout << "Файл создан: " << (PATH_FILES_LAB6 / name) << std::endl;
                file.close();
            } else {
                std::cout << "Не удалось создать файл: " << (PATH_FILES_LAB6 / name)
<< std::endl;
            }
        }
    }
}

```

```

    }
}

void remove(const std::string& name) {
    try {
        if (fs::remove_all(PATH_FILES_LAB6 / name)) {
            std::cout << "Файл или каталог удален: " << name << std::endl;
        } else {
            std::cout << "Файл или каталог не найден: " << name << std::endl;
        }
    } catch (const fs::filesystem_error& e) {
        std::cout << "Ошибка при удалении: " << e.what() << std::endl;
    }
}

void copy(const std::string& source, const std::string& destination) {
    fs::path PATH_FILES_LAB6;
    try {
        fs::copy(PATH_FILES_LAB6 / source, PATH_FILES_LAB6 / destination,
fs::copy_options::recursive);
        std::cout << "Файл или каталог скопирован из " << source << " в " <<
destination << std::endl;
    } catch (const fs::filesystem_error& e) {
        std::cout << "Ошибка при копировании: " << e.what() << std::endl;
    }
}

void rename(const std::string& oldName, const std::string& newName) {
    try {
        fs::rename(PATH_FILES_LAB6 / oldName, PATH_FILES_LAB6 / newName);
        std::cout << "Файл или каталог переименован из " << oldName << " в " <<
newName << std::endl;
    } catch (const fs::filesystem_error& e) {
        std::cout << "Ошибка при переименовании: " << e.what() << std::endl;
    }
}

```

```

#include "Lab4/write_utils.h"
#include <iostream>
#include <vector>
#include <sys/termios.h>

using namespace std;

void start_lab6(int argc, char* argv[]);

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

    start_lab6(argc, argv);
    return 0;
}

void start_lab6(int argc, char* argv[]) {
    if (argc > 1 && std::string(argv[1]) == "/?") {
        printHelp();
        // return 0;
    }

    while (true) {
        printHelp();
        std::cout << "Выберите пункт меню: ";
        int choice;
        std::cin >> choice;
        std::string name1, name2;

        switch (choice) {
            case 1:
                std::cout << "Введите имя файла или каталога для создания: ";
                std::cin >> name1;
                create(name1);
                break;
            case 2:
                std::cout << "Введите имя файла или каталога для удаления: ";
                std::cin >> name1;
                remove(name1);
                break;
            case 3:
                std::cout << "Введите исходное имя и целевое имя для копирования: ";
                std::cin >> name1 >> name2;
                copy(name1, name2);
                break;
            case 4:
                std::cout << "Введите старое и новое имя для переименования: ";
                std::cin >> name1 >> name2;
                rename(name1, name2);
                break;
            case 5:
                std::cout << "Выход из программы.\n";
                return 0;
            default:

```

```
        std::cout << "Неверный выбор, попробуйте снова.\n";  
    }  
}  
}
```

Пример работы:

Использование: TestFileDir Имя1 Имя2

Команды:

- 1 - Создать файл/каталог
- 2 - Удалить файл/каталог
- 3 - Копировать файл/каталог
- 4 - Переименовать файл/каталог
- 5 - Выйти из программы

Для получения справки используйте ключ /?.

Выберите пункт меню: