**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра Вычислительной техники**

**ОТЧЕТ**

**по лабораторной работе №1**

**по дисциплине «Операционные системы»**

**Тема: «Управление файловой системой»**

Студентки гр. 3312 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Половникова А.С.

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Тимофеев А.В.

Санкт-Петербург

2025

**Цель работы.**

исследовать управление файловой системой с

помощью Win32 API.

**Задание 1.1.**

Управление дисками, каталогами и файлами.

Указания к выполнению.

1. Создайте консольное приложение с меню (каждая выполняемая

функция и/или операция должна быть доступна по отдельному пункту меню),

которое выполняет:

∙ вывод списка дисков (функции Win32 API – GetLogicalDrives,

GetLogicalDriveStrings);

∙ для одного из выбранных дисков вывод информации о диске и размер

свободного пространства (функции Win32 API –

GetDriveType, GetVolumeInformation, GetDiskFreeSpace); ∙

создание и удаление заданных каталогов (функции Win32 API –

CreateDirectory, RemoveDirectory);

∙ создание файлов в новых каталогах (функция Win32 API –

CreateFile);

∙ копирование и перемещение файлов между каталогами с возможностью

выявления попытки работы с файлами, имеющими совпадающие

имена (функции Win32 API – CopyFile, MoveFile, MoveFileEx);

∙ анализ и изменение атрибутов файлов (функции Win32 API –

GetFileAttributes, SetFileAttributes, GetFileInformationByHandle,

GetFileTime, SetFileTime).

**Описание кода и работы функций Win32 API**

Программа представляет собой консольное приложение, использующее функции Win32 API для работы с файловой системой. Основные задачи, которые выполняет программа:

* Определение доступных логических дисков.
* Получение информации о дисках.
* Создание и удаление каталогов.
* Работа с файлами: создание, копирование, перемещение.
* Управление атрибутами и временем файлов.

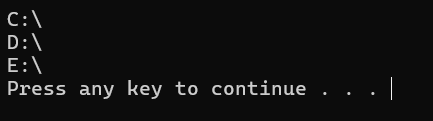
**1. Определение списка логических дисков**

Для получения списка всех доступных логических дисков используется функция *GetLogicalDrives()*. Она возвращает 32-битную битовую маску, где каждый установленный бит соответствует букве диска:

* бит 0 — диск A:
* бит 1 — диск B:
* бит 2 — диск C: и так далее.

Код анализирует эту маску с помощью цикла и битовых операций

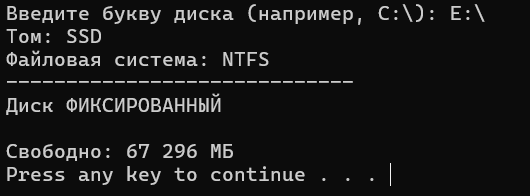
Таким образом, программа выводит все доступные логические диски.



**2. Получение информации о диске**

Функция *ShowDriveInfo()* позволяет получить детальную информацию о выбранном диске. Для этого используются:

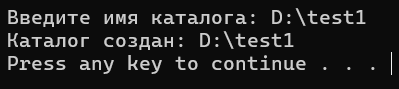
* *GetDriveType()* — определяет тип диска: локальный, съемный, сетевой и т. д.
* *GetVolumeInformation()* — получает имя тома, файловую систему, серийный номер.
* *GetDiskFreeSpaceEx()* — получает объем доступного и свободного пространства.



**3. Создание и удаление каталогов**

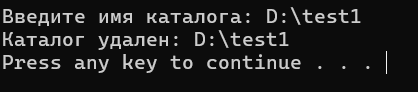
* Для создания папки используется *CreateDirectory()*

При успешном выполнении появится каталог с именем указанном в вызове функции.



* Удаление пустого каталога выполняется с помощью *RemoveDirectory()*:

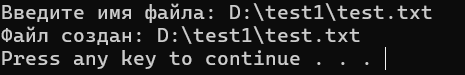
Если папка содержит файлы, она не будет удалена.



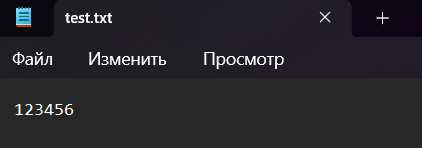
**4. Создание, копирование и перемещение файлов**

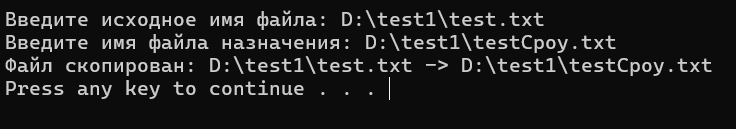
Для работы с файлами программа использует следующие API-функции:

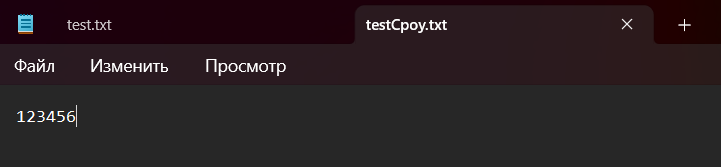
* **Создание файла** с *CreateFile()*



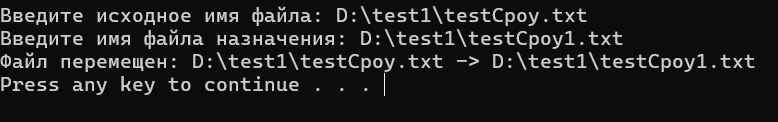
* **Копирование файла** с *CopyFile()*







* **Перемещение или переименование файла** с *MoveFile()*

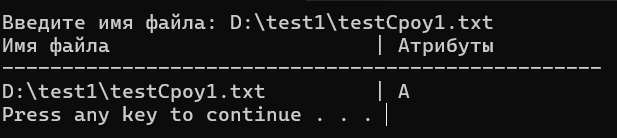




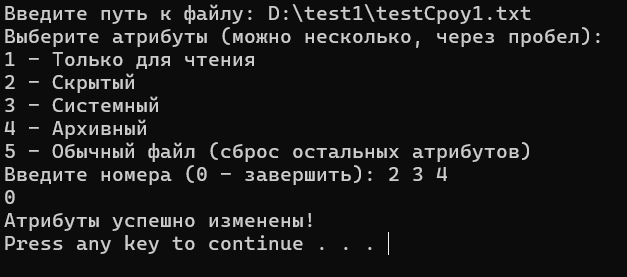
**5. Управление атрибутами файлов**

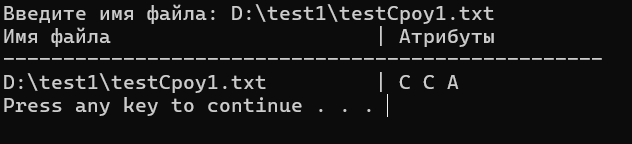
Программа позволяет просматривать и изменять атрибуты файла с помощью *GetFileAttributes()* и *SetFileAttributes()*.

* Получение атрибутов файла *GetFileAttributes()*.



* Установка нового атрибута *SetFileAttributes()*.

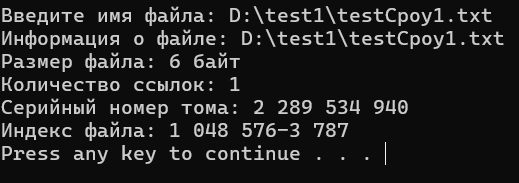




**6. Получение информации о файле**

Для получения информации о файле программа использует *GetFileInformationByHandle()*. Она предоставляет:

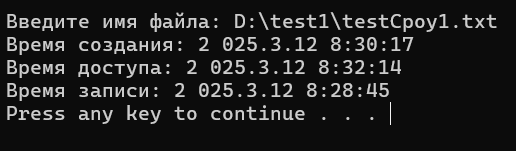
* Размер файла.
* Количество жестких ссылок.
* Серийный номер тома.



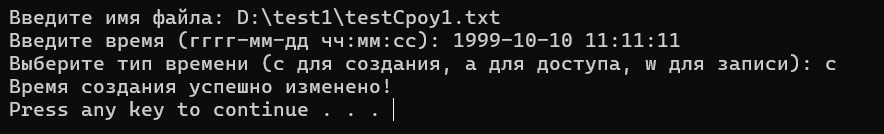
**7. Получение и изменение времени файла**

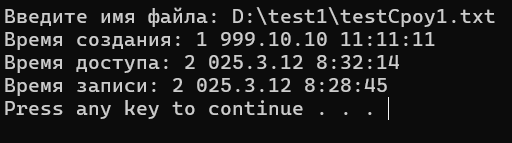
Функции *GetFileTime()* и *SetFileTime()* позволяют просматривать и изменять временные метки файла (создания, последнего доступа, последнего изменения).

* Получение времени *GetFileTime()*.



* Изменение времени *SetFileTime()*.





**8. Работа с пользователем**

Программа предоставляет консольный интерфейс, где пользователь выбирает операции. Ввод осуществляется через *std::wcin*, а вывод — через *std::wcout* для поддержки Unicode.

**Заключение**

Программа демонстрирует основные функции Win32 API для работы с файловой системой. Она позволяет управлять дисками, каталогами и файлами, работать с атрибутами и временем. Использование CreateFile(), GetFileAttributes(), SetFileTime() и других API-функций делает код мощным инструментом для работы с файловой системой Windows.