Лабораторная работа 09

Файловая система

OC, ПОИТ-3

1. ***Внимание! Для работы с файловой системой использовать только OS API.***

**Задание 01.Windows**

1. С помощью Notepad (Notepad+) создайте на дисковом устройстве текстовый файл **OS09\_01.txt.** Заполните его 10 строками из списка студентов вашей подгруппы.



1. Разработайте приложение **OS09\_01**.
2. Приложение **OS09\_01** вызывает функцию **printFileInfo**, имеющую следующий прототип.



1. Функция **printFileInfo** выводит в стандартный поток вывода следующую информацию:

- имя файла;

- тип файла;

- размер файла;

- дата и время создания файла;

- дата и время последнего обновления.

1. Приложение **OS09\_01** вызывает функцию **printFileTxt**, имеющую следующий прототип.



1. Функция **printFileTxt** выводит в стандартный поток вывода содержимое файла.
2. При вызове функции укажите в качестве параметра имя текстового файла, созданного в п.2.
3. Продемонстрируйте работоспособность приложения **OS09\_01**.

**Задание 02.Windows**

1. Разработайте приложение **OS09\_02.**
2. Приложение **OS09\_02** вызывает функцию **delRowFileTxt**, имеющую следующий прототип.



1. Функция применяется к файлу **OS09\_01.txt** (п.2) и вызывается последовательно 4 раза, с row = 1,3,8,10. Результат выполнения продемонстрируйте с помощью функции **printFileTxt** (п.6).
2. Продемонстрируйте работоспособность приложения **OS09\_02**.

**Задание 03.Windows**

1. Разработайте приложение **OS09\_03.**
2. Приложение **OS09\_03** вызывает функцию **insRowFileTxt**, имеющую следующий прототип.



1. Функция применяется к файлу **OS09\_01.txt** (п.2) и вызывается последовательно 4 раза, с row = 0,-1,5,7. Результат выполнения продемонстрируйте с помощью функции **printFileTxt** (п.6).
2. Продемонстрируйте работоспособность приложения **OS09\_03**.

**Задание 04.Windows**

1. Разработайте приложение **OS09\_04.**
2. Приложение **OS09\_04** вызывает функцию **printWathRowFileTxt**, имеющую следующий прототип.



1. Функция применяется к файлу **OS09\_01.txt** (п.2), следит (***используйте функцию наблюдения за файлами в каталоге***) за изменением количества строк в файле в течении mlsec и выводит информацию об изменениях в стандартный поток вывода.
2. Продемонстрируйте работоспособность приложения **OS09\_04** совместно с приложениями **OS09\_03** и **OS09\_04**.

**Задание 05.Linux**

1. Создайте текстовый файл **OS09\_05.txt,** аналогичный файлу **OS09\_01.txt** (п.2).
2. Разработайте приложение **OS09\_05,** подсчитывающее количество строк и выводящее это значение в стандартный поток.
3. Продемонстрируйте работоспособность приложения **OS09\_05**.

**Задание 06.Linux**

1. Разработайте приложение **OS09\_06,** принимающее 1 параметр, принимающее целочисленное числовое значение.
2. Если числовое значение принимает нечетное значение, то приложение создает новый файл **OS09\_06\_1.txt**, содержащий только нечетные строки из файла **OS09\_05.txt**.
3. Если числовое значение принимает четное значение, то приложение создает новый файл **OS09\_06\_2.txt**, содержащий только четные строки из файла **OS09\_05.txt**.
4. Продемонстрируйте работоспособность приложения **OS09\_06**.

**Задание 07.Linux**

1. Разработайте приложение **OS09\_07,** демонстрирующее возможности функции **lseek.**
2. Поясните назначение и принцип работы функции **lseek**.

**Задание 08.** Ответьте на следующие вопросы

1. Что такое файл?
2. Перечислите основные характеристики (атрибуты) файла.
3. Что такое файловая система?
4. Перечислите основные функции файловой системы.
5. Перечислите 3 названия файловой системы.
6. Какая файловая система установлена на вашем компьютере под Windows? под Linux?
7. Что такое каталог файловой системы? перечислите наименования специальных каталогов.
8. Поясните понятие «текущий каталог приложения».
9. Что такое специальные имена файлов? перечислите их, для чего они нужны.
10. Для чего используются буферы ввода-вывода?
11. Поясните понятие «кэширование».
12. Поясните понятие «указатель позиции файла».
13. Поясните понятие «маркер конца файла».
14. Поясните понятие «блокировка файла».
15. Windows. Функция OS API для создания файла.
16. Windows. Функция OS API для открытия файла.
17. Windows. Функция OS API для удаления файла.
18. Windows. Функция OS API для записи в файл.
19. Windows. Функция OS API для чтения файла.
20. Windows. Назначение и отличие функций OS API: CopyFile, MoveFile, ReplaseFile.
21. Windows. Перечислите функции OS API, которые изменяют текущее значение указателя позиции файла.
22. Windows. Перечислите функции OS API для блокировки и разблокировки файлов.
23. Windows. Поясните механизм «наблюдение за каталогом», перечислите набор функций OS API, позволяющий реализовать этот механизм.
24. Windows. Перечислите функции OS API для работы с каталогами, поясните их назанчения.
25. Linux. Что такое FHS?
26. Linux. Перечислите типы файловых систем.
27. Linux. Что такое inode?
28. Linux. Поясните назначение функций open, read, write, close, ioctl, stat, flush**,** lseek, lstat, fstat.

# 1. Что такое файл?

Абстракция для унифицированного доступа к данным; набор логических записей

# 2. Перечислите основные характеристики (атрибуты) файла.

Тип, расположение, размер, дата и время (создания, изменения и открытия)

# 3. Что такое файловая система?

Часть операционной системы, обеспечивающая доступ к файлам

# 4. Перечислите основные функции файловой системы.

* создание/удаление каталогов,
* включение/исключение подкаталогов,
* включение/исключение файла в каталог,
* создание/удаление файла,
* открытие/закрытие доступа к файлу,
* чтение/запись логических записей файла,
* установка (поддержка) указателя файла;

# 5. Перечислите 3 названия файловой системы.

FAT NTFS ReFS

# 6. Какая файловая система установлена на вашем компьютере под Windows? под Linux?

NTFS,

# 7. Что такое каталог файловой системы? перечислите наименования специальных каталогов.

Файл, содержащий информацию о месте расположения других файлов; точка, две точки

1. **C:\Windows:** Системные файлы операционной системы Windows.
2. **C:\Program Files:** Установленные программы по умолчанию.
3. **C:\Users:** Домашние каталоги пользователей, содержащие их личные файлы и настройки.
4. **C:\ProgramData:** Общие данные программ для всех пользователей.
5. **C:\Documents and Settings:** Домашние каталоги пользователей в старых версиях Windows.

Для операционной системы Linux:

1. **/bin:** Бинарные (исполняемые) файлы и команды системы.
2. **/etc:** Конфигурационные файлы системы.
3. **/home:** Домашние каталоги пользователей.
4. **/var:** Переменные данные, такие как журналы (логи), временные файлы и кэши.
5. **/usr:** Системные файлы, включая программы, библиотеки и документацию.
6. **/tmp:** Временные файлы.

# 8. Поясните понятие «текущий каталог приложения».

Текущим называется каталог, с которым работает ОС, если ей не указать другого каталога. Он обозначается точкой (.)

# 9. Что такое специальные имена файлов? перечислите их, для чего они нужны.

Имена, которые не могут быть именами файлов: con, lpt1, prn, aur, com

1. **CON:** Отсылается к консоли и обозначает устройство ввода-вывода.
2. **LPT1:** Первый порт принтера.
3. **PRN:** Обозначение принтера.
4. **AUX:** Вторичное устройство, как правило, используемое для подключения звуковых устройств.
5. **COM1, COM2, COM3, и так далее:** Серийные порты.

# 10. Для чего используются буферы ввода-вывода?

Используются для устранения несоответствия между физическим и логическим чтением/записью

# 11. Поясните понятие «кэширование».

Перемещение в быстродействующую память

# 12. Поясните понятие «указатель позиции файла».

Объект файловой системы, позиционирующий логическую запись

# 13. Поясните понятие «маркер конца файла».

является индикатором операционной системы, означающим, что данные в источнике закончились

# 14. Поясните понятие «блокировка файла».

Механизм, который ограничивает доступ к файлу, давая доступ в данный момент только одному пользователю или процессу

# 15. Windows. Функция OS API для создания файла.

CreateFile с параметром Create\_New

# 16. Windows. Функция OS API для открытия файла.

CreateFile с параметром Open\_Always

# 17. Windows. Функция OS API для удаления файла.

DeleteFile

# 18. Windows. Функция OS API для записи в файл.

WriteFile

# 19. Windows. Функция OS API для чтения файла.

ReadFile

# 20. Windows. Назначение и отличие функций OS API: CopyFile, MoveFile, ReplaseFile.

Copy - копирует,

Move - перемещате,

Replase - замещает в рамках диска

# 21. Windows. Перечислите функции OS API, которые изменяют текущее значение указателя позиции файла.

SetFilePointer,

SetFilePointerEx

# 22. Windows. Перечислите функции OS API для блокировки и разблокировки файлов.

LockFile, UnlockFile

# 23. Windows. Поясните механизм «наблюдение за каталогом», перечислите набор функций OS API, позволяющий реализовать этот механизм.

В Windows для реализации механизма "наблюдение за каталогом" используется API, который предоставляет функционал для мониторинга изменений в файловой системе. Основные функции, которые используются для этого, входят в состав API Windows:

FindFirstChangeNotification,

FindNextChangeNotification,

FindCloseChangeNotification,

WaitForSingleObject

# 24. Windows. Перечислите функции OS API для работы с каталогами, поясните их назанчения.

CreateDirectory - Создает новый каталог,

FindFirstFile - Ищет файл или каталог по указанному пути с использованием шаблона,

RemoveDirectory - Удаляет указанный каталог,

MoveFile - Перемещает файл или каталог в новое местоположение,

GetCurrentDirectory - Получает текущий рабочий каталог для текущего процесса

# 25. Linux. Что такое FHS?

Стандарт иерархии файловой системы

# 26. Linux. Перечислите типы файловых систем.

Ext2, Ext3, Ext4, JFS, ReiserFS, XFS, Drtfs, ZFS

# 27. Linux. Что такое inode?

структура данных в которой хранится информация о файле или директории в файловой системе.

# 28. Linux. Поясните назначение функций open, read, write, close, ioctl, stat, flush, lseek, lstat, fstat.

**open** - открыть файл

**read** - чтение из файла

**write** - запись в файл

**close** - закрыть файл

**ioctl** - управляет аппаратными устройствами

**stat** - состояние файла

**flush** - Синхронизирует связанный буфер потока с его управляемой выходной последовательностью

**lseek** - устанавливает указатель положения в файле

**lstat** - считывание состояние файла

**fstat** - заполняет структуру информацией о файле