**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИИ**  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ  
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В.Г.ШУХОВА»**  
**(БГТУ им. В.Г.Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №5

Операционные системы

Операции с файлами

Выполнил: ст. группы ВТ-31  
Ковалёв И. Д.

Проверил: Михелев В. М.

**Белгород 2019**

### Файлы инициализации.

Файл инициализации - специальный текстовый файл формата \*.ini, который используется при загрузке среды и отдельных программ для установки начальных параметров.

Также используется windows для сохранения системных настроек, а также приложениями для сохранения их собственных данных инициализации. Windows позволяет каждому приложению создавать и поддерживать собственные файлы инициализации.

Имеют стандартный формат:

Каждый файл делится на секции. У каждой секции есть имя, кот. указано в файле в квадратных скобках. Секция содержит записи, которые состоят из двух частей, разделенных знаком =. Первая часть - имя ключа, вторая часть - строковое значение.

В Windows NT большинство секций WIN.INI и других системных файлов

инициализации отображается на реестр. Если секция INI-файла отображается на реестр, функции работы с INI-файлами записывают данные прямо в реестр, и эти данные вообще не попадают в файлы инициализации.

Системный реестр.

Системный реестр - иерархическая централизованная база данных, используемая для хранения данных, необходимых для правильного функционирования Windows. В нем хранятся такие данные, как профили пользователей, сведения об установленном программном обеспечении и типах файлов, а также информация об используемом оборудовании.

Имеет иерархическую структуру, состоящую из различных разделов, которые называются ключами, которые, в свою очередь, содержат параметры или вложенные ключи.

Существует 6 корневых ключей:

HKEY\_CURRENT\_USER - Данные конфигурации текущего пользователя. Является ссылкой на один из вложенных разделов HKEY\_USERS.

HKEY\_USERS - Профили всех пользователей ОС.

HKEY\_LOCAL\_MACHINE - Параметры конфигурации, относящиеся к данному компьютеру.

HKEY\_CLASSES\_ROOT - ассоциации между приложениями и типами файлов.

HKEY\_CURRENT\_CONFIG - сведения об оборудовании, используемом компьютером при запуске операционной системы.

HKEY\_PERFORMANCE\_DATA - показания счетчиков производительности. Информацию оттуда нельзя получить через regedit, но можно с помощью других приложений или api-функций из winreg.h

Параметры ключа могут иметь значения таких типов:

REG\_BINARY - двоичный параметр.

REG\_DWORD - аналог DWORD, без знака.

REG\_EXPAND\_SZ - строка переменной длины.

REG\_MULTI\_SZ - многострочный параметр.

REG\_QWORD - аналог QWORD, без знака

REG\_SZ - строка фиксированной длины.

Некоторые api-функции для работы с реестром:

RegCIoseKey - Закрывает открытый ключ

RegCreateKey,RegCreateKeyEx - Создает новый ключ

RegDeleteKey - Удаляет ключ

RegDeleteValue - Удаляет параметр

RegEnumKey, RegEnumKeyEx - Перебирает подключи заданного ключа

RegEnumValue - Перебирает параметры ключа

RegOpenKey,RegOpenKeyEx - Открывает ключ

RegQueryInfoKey - Получает информацию о ключе

RegQueryValue,RegQueryValueEx - Читает значение параметра

Асинхронные файловые операции.

В win32 есть возможность выполнять файловые операции в асинхронном режиме - выход из функции пересылки данных происходит немедленно, а операционная система завершает операцию в фоновом режиме.

Для этого используется структура OVERLAPPED:

typedef struct \_OVERLAPPED {

//Не меняются пользователем, исп. для внутренней работы

ULONG\_PTR Internal; ULONG\_PTR InternalHigh;

DWORD Offset; //Смещение в файле, в котором происходит операция

DWORD OffsetHigh;//Смещение в файле, в котором происходит операция

HANDLE hEvent; //Манипулятор события

} OVERLAPPED;

Манипулятор события - объект особого типа, который позволяет определить момент заверщения перекрывающейся операции.

Файловое время.

В windows сущетсвует механизм регистрации времени и даты создания файла и последнего обращения к нему. Для хранения используется 64-разрядная величина, содержащая количество 100-наносекундных интервалов с 1 января 1601 года.

typedef struct \_FILETIME {

DWORD dwLowDateTime;

DWORD dwHighDateTime;

} FILETIME, \*PFILETIME;

Файловове время хранится в системе в формате UTC, но ничего не мешает использовать api-функции для преобразования в местное время.

На каждую часть времени выделяется 4 бита (день/месяц/год/час/минута/секунда).

Некоторые функции для работы с файловым временем:

CompareFileTime Сравнивает две структуры FILETIME

DosDateTimeToFileTime - Преобразует файловое время DOS в FILETIME

FileTimeToDosDateTime - Преобразует FILETIME в файловое время DOS

FlleTimeToLocalFileTime - Преобразует структуру FILETIME из UTC в местное время

FileTimeToSystemTime - Заполняет структуру SYSTEMTIME временем, заданным

структурой FILETIME

GetFileTime - Получает время создания, обращения и модификации файла

SetFileTime - Задает время создания, обращения и модификации файла

SystemTimeToFileTime - Заполняет структуру FILETIME временем, заданным

структурой SYSTEMTIME

Функций для работы с файлами и каталогами.

CopyFile, CopyFileEx - Копирует файл

CreateDirectory,CreateDirectoryEx - Создает новый каталог

CreateFile - Создает или открывает файл

DeleteFile - Удаляет файл

DeviceIoControl - Поддерживает низкоуровневые операции с дисковыми

томами (и другие операции с устройствами)

GetBinaryType - Проверяет, является ли файл исполняемым

GetCompressedFileSize Получает фактический объем, занимаемый сжатым

файлом на диске

GetCurrentDirectory  - Получает текущий каталог

GetDiskFreeSpace - Получает объем свободного места на диске

**ВАЖНО!** GetDiskFreeSpace возвращает не количество свободных байт, а количество свободных кластеров

GetDriveType - Получает тип заданного дискового устройства

GetFileAttributes - Получает атрибуты файла

GetFileInformationByHandle - Получает информацию о файле

GetFileSize, GetFileSizeEx - Получает размер файла

GetFileType - Получает тип файла по манипулятору файла

GetFullPathName - Получает полное имя файла

GetLogicaIDrives,GelLogicalDriveStrings - Получает список логических дисков, определенных в системе

GetOvarlappedResult - Получает результат асинхронной файловой операции

GetShortPathName - Получает короткое имя файла

GetSystemDirectory - Получает путь к системному каталогу Windows

GetTempFileName - Получает уникальное имя временного файла

GetTempPath - Получает путь к каталогу, выделенному под временные

файлы

GetVolumelnformation - Получает информацию о дисковом томе

GetWindowsDirectory - Получает путь к каталогу Windows

SetFilePointer, SetFilePointerEx - Переходит к заданной позиции в файле

LockFile, LockFileEx - Блокирует часть файла

Movefile, MoveFileEx - Перемещает файл

ReadFile, ReadFileEx - Читает данные из файла

WriteFile, WriteFileEx - Записывает данные в файл

QueryDosDevice - Получает информацию об именах логических устройств

DOS, присутствующих в системе

RemoveDirectory - Удаляет каталог

SearchPath - Ищет файл, используя стандартную последовательность

поиска Windows

SetCurrentDirectory - Задает текущий каталог

**Цель работы:** Получение практических навыков по использованию Win32 API для исследования памяти Windows.

**Задания к работе:**

Реализовать программу, считывающую из системного реестра данные о системных службах Windows HKLM\SYSTEM\ \CurrentControlSet\Services). Обеспечить возможность изменения описания службы и способа запуска. Информацию о службах выводить в файл.