**37. Типы файлов.**

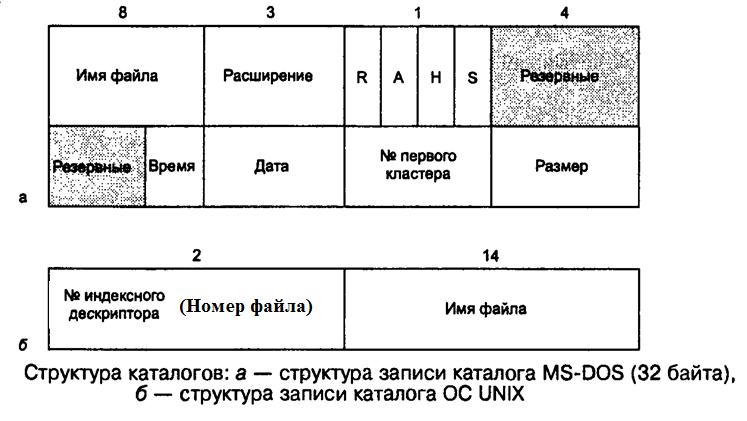
**Основные типы файлов: (5 типов)**  
1. Обычные файлы - содержат информацию пользователя:

* текстовые файлы (например ASCII и Unicode)
* двоичные файлы, не являются текстовыми (бинарные, архивные, исполняемые).

2. Каталоги —особый системный тип файлов, содержащий информацию о наборе файлов, сгруппированных пользователем по определенному признаку.

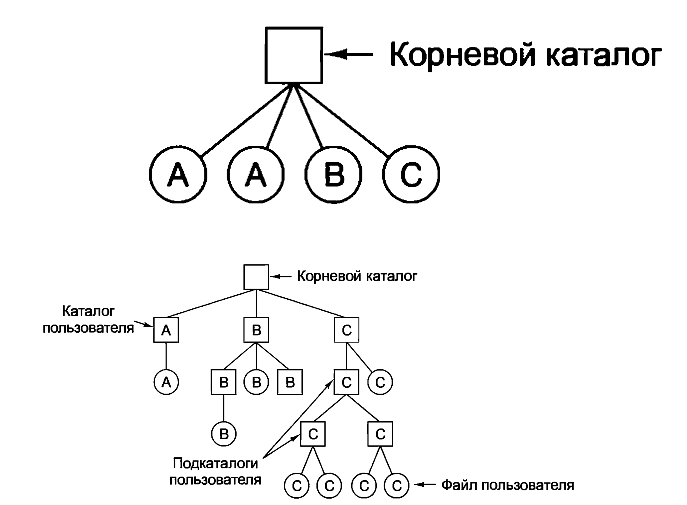
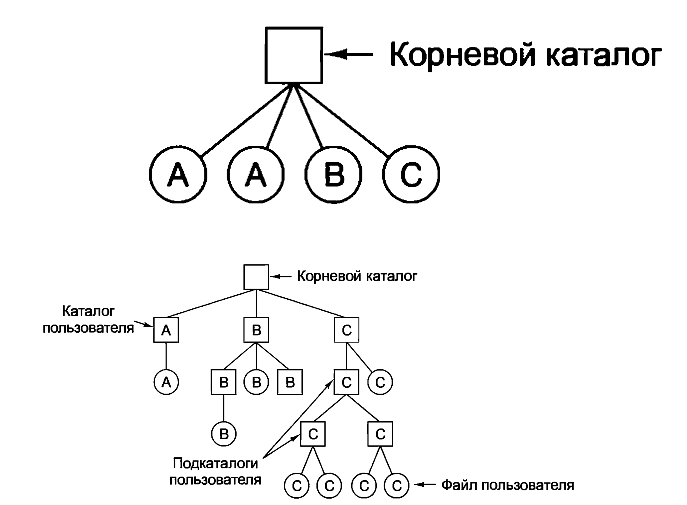
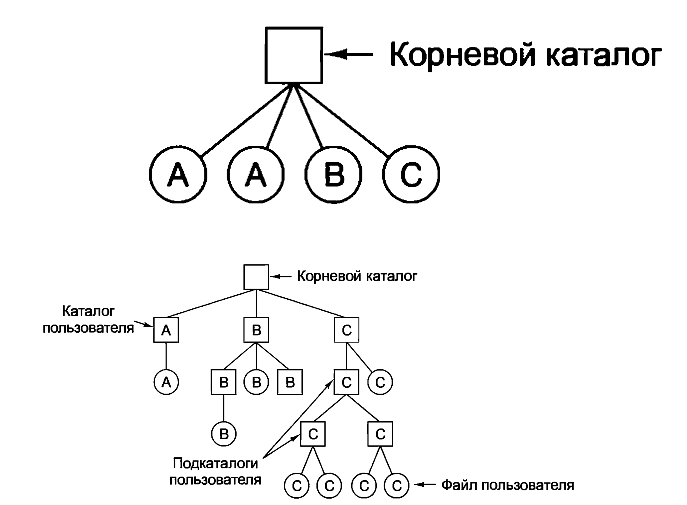
*Примеры структуры каталога*  
Значения атрибутов могут размешаться:

1) непосредственно в каталогах, в которых хранятся файлы (FAT),

2) в каталогах хранятся только ссылки на таблицы, в которых хранятся атрибуты (NTFS) и имя файла.

R-только для чтения A- архивный H- скрытый S-системный

*Структура каталогов*

 3. Специальные файлы — это фиктивные файлы, связанные с устройствами ввода-вывода

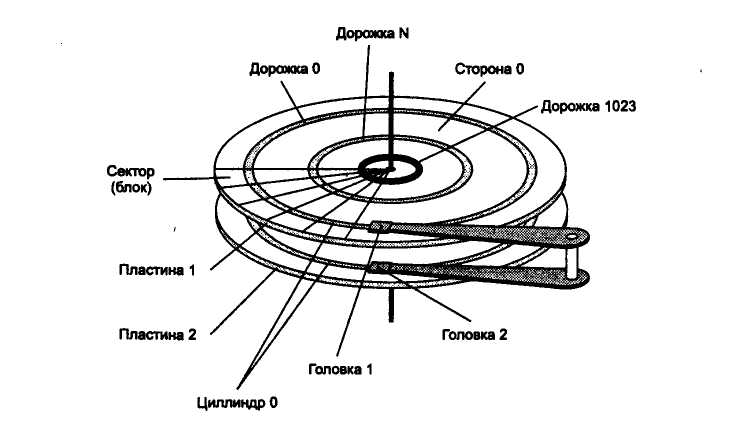
Специальные файлы позволяют выполнять операции ввода-вывода в устройство с помощью обычных команд записи в файл или чтения из файла.

* Символьные специальные файлы это фиктивные файлы связанные с «байтовыми» устройствами ввода/вывода и используются для организации одинакового механизма доступа как к файлам, так и внешним устройствам (USB, COM).
* Блочные специальные файлы, связанные с «блочными» устройствами ввода-вывода и используются для организации доступа к дисковой памяти.

4. Файлы символическая связь (символическая ссылка - ярлык)

5. Именованный канал (канал FIFO «первым вошел – первым вышел») Программный канал – это файл особого типа. Процессы могут записывать и считывать данные из канала как из обычного файла.)

**38. Физическое размещение файлов на диске(понятие дорожки, цилиндра, сектора, низкоуровневое форматирование). Назначение низкоуровневого и высокоуровневого форматирования**



На каждой стороне каждой пластины размечены тонкие концентрические кольца — дорожкы (traks), на которых хранятся данные.Нумерация дорожек начинается с 0 от внешнего края к центру диска.

Совокупность дорожек одного радиуса на всех поверхностях всех пластин пакета называется цилиндром (cylinder).

Каждая дорожка разбивается на фрагменты, называемые секторами (sectors), или блоками (blocks).

Все дорожки имеют равное число секторов. Сектор имеет фиксированный для конкретной системы размер, кратный степени двойки (чаще всего 512 байт).

Сектор — наименьшая адресуемая единица обмена данными дискового устройства с оперативной памятью.

Адрес сектора состоит из : номер цилиндра + номер поверхности (головки) + номер сектора.  
  
**Низкоуровневое форматирование диска**

Процесс разметки диска на дорожки и сектора называется низкоуровневым форматированием диска (как правило на заводе изготовителе).

Низкоуровневое форматирование диска не зависит от типа операционной системы, которая этот диск будет использовать.

**39. Понятие кластера, CHS, LBA и связь между ними. Высокоуровневое форматирование.**

Кластер (cluster) – единица данных, которую использует ОС для обращения к диску. Кластер состоит из набора секторов. При создании файла место на диске ему выделяется кластерами.

Высокоуровневым форматирование - процесс назначения группам физических секторов диска номеров кластеров и создания таблиц кластеров и структур данных под конкретный тип файловой системы.

При высокоуровневом форматировании определяется размер кластера и на диск записывается информация, необходимая для работы файловой системы:

- доступное пространство,

- информация о поврежденных областях.

**CHS, LBA (Сразу прочитать про разделы)**  
Адреса секторов разделов задаются:

* либо в формате CHS (Cylinder-Head-Sector — Цилиндр-Головка-Сектор), например 6 - 1 – 32.
* либо LBA формате(Logical Block Address – Логический Адрес Блока, в виде последовательности номеров 01 2 ….).

Конкретный формат определяется типом раздела (Boot ID), записанным в 04h байте  
  
**CHS**  
CHS устаревший вариант, использовался для гибких дисков

ОС при обращении к диску должна учитывать геометрию диска (количество цилиндров, головок, секторов)

Вид адресации CHS или LBA задается в BIOS

Из – за ограниченного количества разрядов (24) заложенных разработчиками первых BIOS для хранения значений CHS при размере сектора 512 байт позволял поддерживать диски не более 8Гб

224 \*512 = 8 Гб

Диски более 8Гб были не видны в системе  
**LBA**

Каждый логический блок, адресуемый на жёстком диске, имеет свой номер - целое число, начиная с нуля LBA=0, LBA=1, ...)

LBA 0 = Цилиндр 0/Головка 0/Сектор 1

В настоящее время для задания номера блока используется 48 бит (возможно 64), что даёт возможность адресовать на приводе (248) 281 474 976 710 656 блоков (при блоке в 512 байт, 128 ПИБайт). 1ПИБайт = 250 байт  


C- номер цилиндра,

h - номер головки,

s - номер сектора,

H - число головок,

S - число секторов на дорожке

**40. Понятие основного, дополнительного и активного раздела, главной загрузочной записи и таблицы разделов, загрузочной записи раздела. Последовательность загрузки ОС.**

Прежде чем форматировать диск под определенную файловую систему, он может быть разбит на разделы

Раздел — это непрерывная часть физического диска, которую операционная система представляет пользователю как логический диск.

На каждом логическом диске может создаваться только одна файловая система.

На разных логических дисках одного и того же физического диска могут располагаться файловые системы разного типа.

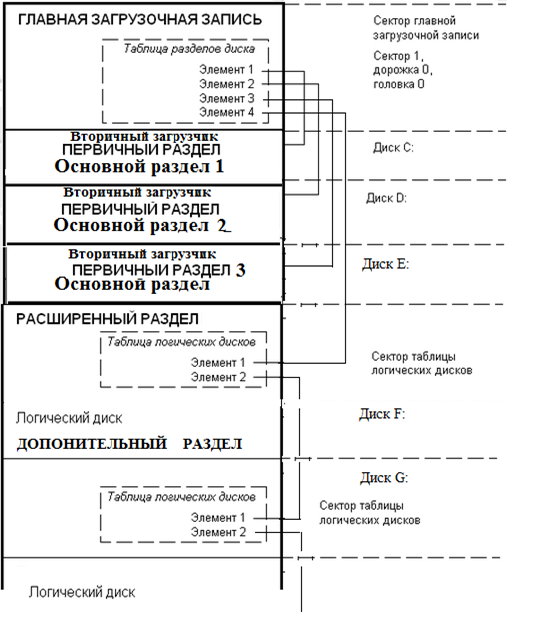
Все разделы одного диска имеют одинаковый размер секторов.

В результате высокоуровневого форматирования размеры кластеров могут быть различными для различных разделов.

Том - логическое устройство, отформатированное под конкретную файловую систему.

* Раздел, на который может быть установлена ОС называется основным (первичным).
* Основной раздел, на который установлена ОС называется активным.
* Раздел, который не является основным, называется дополнительным

**Разбивка диска на разделы**

МBR – главная загрузочная  
запись (первичный загрузчик)   
Создается при разбивке диска на разделы. PBR ( Partition Boot Record - загрузочная запись раздела) Создается при высокоуровневом форматировании на каждом 

* Сектор 0 на нулевой дорожке называется главной загрузочной записью - MBR(Master Boot Record – первичный загрузчик не зависящий от ОС)
* В конце MBR содержится таблица разделов. Из этой таблицы берутся начальные и конечные адреса каждого раздела.
* Раздел, на котором установлена ОС помечается в этой таблице активным.
* В каждом разделе диска при высокоуровневом форматировании создается загрузочный блок PBR ( Partition Boot Record ) , даже если он не содержит загружаемой операционной системы. В будущем он может содержать какую-нибудь операционную систему.
* При установке ОС в загрузочный блок записывается загрузчик операционной системы (вторичный загрузчик).

**Последовательность загрузки системы**

* Самотестирование при включении питания.
* Обнаружение и инициализация видеокарты.
* Отображение стартового экрана BIOS.
* Осуществление быстрой проверки памяти (RAM).
* Конфигурация устройств plug and play.
* Определение загрузочного устройства.
* При загрузке компьютера BIOS (базовая система ввода-вывода) считывает и загружает MBR. Программа MBR, — находит расположение активного раздела, считывает в память его первый блок , содержащий вторичный загрузчик.
* Вторичный загрузчик вызывает загрузчик ОС, который загружает операционную систему, содержащуюся в этом разделе.

**41. Физическое размещение файлов связанными списками. Структура каталога FAT. Организация файловой системы FAT.**

Основными критериями эффективности физической организации файлов являются:

□ скорость доступа к данным;

□ объем адресной информации файла;

□ степень фрагментированности дискового пространства;

□ максимально возможный размер файла.

Существуют следующие способы организации физического размещения файлов:

- непрерывное размещение;

- размещение в виде связанного списка

- размещение с помощью i – узлов(node)

**Размещение связанным списком кластеров**

* В начале каждого кластера содержится указатель на следующий кластер.
* Адресная информация минимальна: расположение файла может быть задано одним числом — номером первого кластера, который хранится в записи информации о файле в каталоге. Все остальные кластеры файла могут быть найдены начиная с первого.

Достоинства:

- Для размещения файла может использоваться любой свободный кластер, который может быть подсоединен к цепочке.

- Внешняя фрагментация отсутствует.

- Файл может изменять свой размер во время своего существования.

Недостаток:

- Сложность реализации доступа к произвольному кластеру файла. необходимо прочитать все файлы до искомого (медленно);

- Указатели на кластеры занимают значительный объем.

**Размещение связанным списком индексов**

Файлу также выделяется память в виде связанного списка кластеров.

С каждым кластером диска связывается некоторый элемент — индекс. Индексы располагаются в отдельной области диска FAT (File Allocation Table), занимающая один кластер

Достоинства:

-Работаем с таблицей индексов, а не с кластерами.

- Для быстроты поиска таблицу можно загрузить в память.

- С каждого кластера убирается указатель (размер кратен 2)

Недостаток- большие таблицы файлов и как следствие медленный поиск.

**Характеристики FAT 32**

В зависимости от количества бит, используемых для хранения номера следующего кластера файловые системы бывают:

FAT – 16

FAT – 32

FAT32 поддерживает размеры кластеров 512, 1024, 2048, 4096, 8192, 16КБ, 32КБ, 64КБ (по умолчанию 32 Кб)

Поддерживаемый размер диска зависит от размера кластера. Для кластера 32 КБ приблизительно равен 8 терабайт.

Максимальный размер файла FAT32 - не более 4 ГБ.

Максимальное количество файлов - не более 268 435 444

**Структура диска при FAT**

Загрузочный сектор, содержащий загрузочную запись раздела (boot record);

Таблицы размещения файлов (FAT, File Allocation Table);

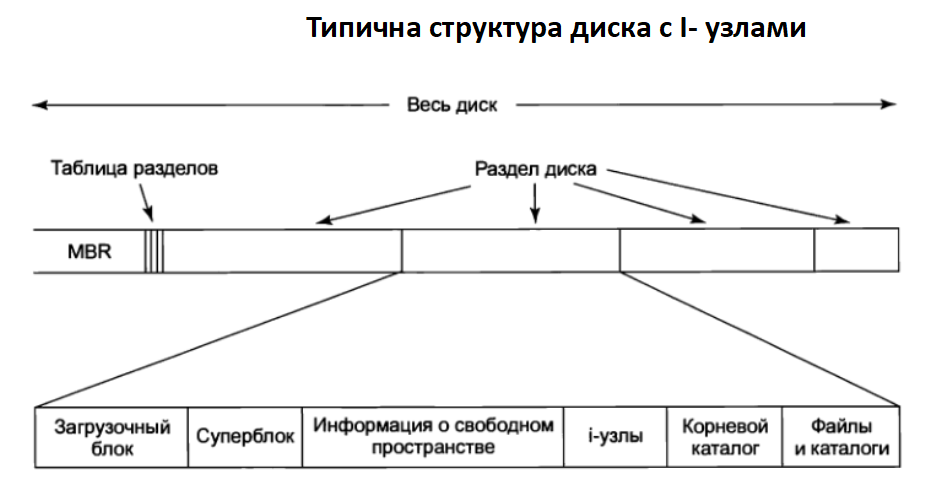
Корневой каталог (Root directory, ROOT).  
  
**42. Физическое размещение файлов с помощью I-узлов, структура каталога.**

I-узлы ((index-node — индекс-узел),

* Каждому файлу ставиться в соответствие структуры данных,
* Называемой i-узлом содержащей атрибуты файла и дисковые адреса его блоков, заданные простым перечислением адресов всех кластеров файла.
* При использовании i-узла появляется возможность найти все блоки файла.
* Преимущество над связанными списками - в памяти надо хранить не всю таблицу а только i-узел открытого файла.

Достоинства: Высокая скорость поиска.

Недостаток: Длина адреса зависит от размера файла. Для больших файлов может быть велика. Для уменьшения адреса прямой способ адресации сочетается с косвенным.



Для уменьшения адреса прямой способ адресации сочетается с косвенным. Дополнительные блоки адресов хранятся в дополнительных записях на жестком диске.

