

## Файлы и каталоги

Из курса информатики основной школы вам известно, что **файл** — это поименованная совокупность данных определённого размера, размещаемая на внешних устройствах (носителях информации) и рассматриваемая в процессе обработки как единое целое.

Файл характеризуется набором параметров (имя, размер, дата создания, дата последней модификации) и атрибутами, используемыми операционной системой для его обработки (архивный, системный, скрытый, только для чтения). Размер файла выражается в байтах.

На каждом компьютерном носителе информации (жёстком, оптическом диске или флеш-памяти) может храниться большое количество файлов. Для удобства поиска информации файлы по определённым признакам объединяют в группы, называемые каталогами или папками.

**Каталог (папка)** — это поименованная совокупность файлов и подкаталогов (вложенных каталогов). Можно сказать, что каталог — это тоже файл, но только содержащий в себе информацию о заключённых в него файлах. Правда папки можно создавать средствами операционной системы, а вот для создания большинства типов файлов понадобится использование прикладного программного обеспечения.

## Функции файловой системы

Работу пользователя с файлами обеспечивает подсистема ОС, называемая файловой системой.



**Файловая система** — это часть ОС, определяющая способ организации, хранения и именования данных на носителе информации.

Современные файловые системы решают следующие задачи:

- определяют правила построения имён файлов и каталогов;
- поддерживают программный интерфейс работы с файлами для приложений (предоставляют приложениям функции для работы с файлами);
- определяют порядок размещения файлов на диске;
- обеспечивают защиту данных в случае сбоев и ошибок;
- обеспечивают установку прав доступа к данным для каждого конкретного пользователя;
- обеспечивают совместную работу с файлами (если один пользователь открыл файл, то для остальных устанавливается режим «только чтение»).

Рассмотрим некоторые из этих функций более подробно.

## Правила построения имён файлов и каталогов

Файловые системы современных ОС допускают использование имён длиной до 255 символов, причём в них можно использовать буквы национальных алфавитов и пробелы.

В ОС Windows в имени файла запрещено использование следующих символов: \, /, \*, ?, ", <, >, |. В Linux эти символы, кроме /, допустимы, хотя использовать их следует с осторожностью, т. к. некоторые из них могут иметь специальный смысл, а также из соображений совместимости с другими ОС.

Операционная система Linux, в отличие от Windows, различает строчные и прописные буквы в имени файла, например: FILE.txt, file.txt и FiLe.txt — это в Linux три разных файла.

## Порядок размещения файлов на диске

Файл, представляемый нами как единое целое, на самом деле может быть разбросан «кусочками» по всему диску. Минимальный размер такого «кусочка» (кластера, блока) — от 512 байт до 64 Кбайт в зависимости от используемой файловой системы. При размещении на диске каждому файлу отводится целое число кластеров. Например, на файл размером 65 Кбайт будет отведено два кластера по 64 Кбайта, при этом второй кластер будет считаться занятым, хотя фактически значительная его часть использоваться не будет.



Использование кластеров больших размеров обеспечивает следующие преимущества: 1) повышается скорость чтения/записи файлов, имеющих большие размеры; 2) увеличивается максимальный объём диска, который поддерживает файловая система.

Как вы можете это объяснить?

Почему в файловой системе с большими кластерами (блоками) невыгодно хранить маленькие файлы? К чему это может привести?

## Защита данных в случае сбоя и ошибок

Эта функция обеспечивается за счёт журналирования, суть которого состоит в следующем. Перед началом выполнения операций с файлами ОС записывает (сохраняет) список действий, которые она будет проводить с файловой системой. Эти записи хранятся в отдельной части файловой системы, называемой журналом. Как только изменения файловой системы внесены в журнал, она применяет эти изменения к файлам, после чего удаляет эти записи из журнала. Если во время выполнения операций с файлами произошёл сбой, то по записям в журнале можно определить пострадавшие файлы и восстановить их.

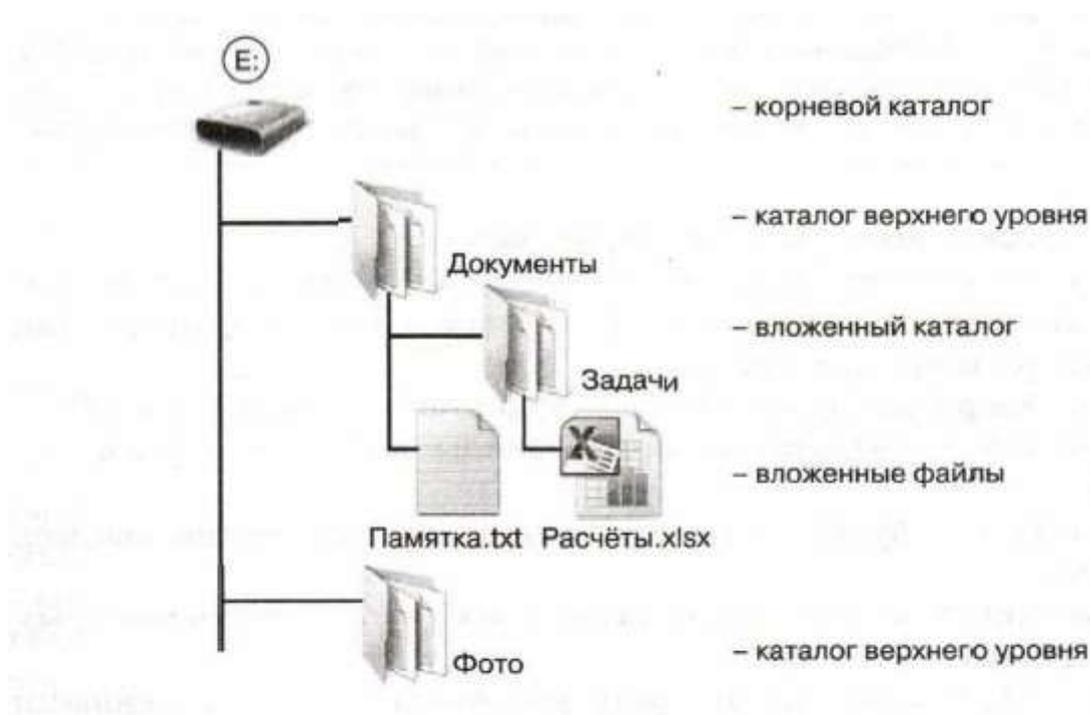
# Файловые структуры

Совокупность файлов на диске и взаимосвязей между ними называют **файловой структурой диска**. Первоначально файловые системы поддерживали только одноуровневые файловые структуры: все файлы хранились в одном каталоге.

Для хранения большого (сотни и тысячи) количества файлов используются иерархические (многоуровневые) файловые структуры: файлы группируются в каталоги, каталоги могут группироваться в каталоги более высокого уровня.

Пользователь, объединяя по собственному усмотрению файлы в каталоги, может создать удобную для себя систему хранения информации.

Графическое изображение иерархической файловой структуры называется деревом. Древоподобные иерархические структуры можно изображать вертикально и горизонтально (рис. 2.10).



**Рис. 2.10.** Типовая древоподобная иерархическая структура, принятая в Windows

Чтобы обратиться к нужному файлу (каталогу), хранящемуся на некотором диске, можно указать путь (адрес каталога) — набор символов, показывающий расположение файла в файловой системе.

Полное имя файла — запись пути к файлу, завершаемая именем файла.

Например:

E:\Документы\Задачи — путь к файлу Расчёты.xlsx;

E:\Документы\Задачи\Расчёты.xlsx — полное имя файла Расчёты, .xlsx.

Для того чтобы «добраться» до нужного файла в многоуровневой файловой структуре, пользователю совсем не обязательно перемещаться по каталогам, образующим путь к файлу. Современные операционные системы имеют специальные инструменты, позволяющие достаточно быстро находить нужные файлы даже в том случае, когда точно не известно их расположение. Если пользователь знает только некоторую часть имени файла (например, он знает, что в имени файла обязательно должна быть цифра 7), то для его поиска можно воспользоваться маской имени файла.



Маска имени файла — последовательность букв, цифр и других допустимых в именах файлов символов, а также символов «?» и «\*», определяющая те или иные требования к имени файла.

Символ «?» (вопросительный знак) означает, что на его месте в имени файла должен быть ровно один произвольный символ. Символ «\*» (звёздочка) означает, что на его месте в имени файла может быть последовательность любых символов произвольной длины, в том числе «\*» может задавать и пустую последовательность.

Рассмотрим несколько примеров масок:

- маске **\*.\*** соответствуют абсолютно все файлы, поскольку набор произвольных символов указывается как для имени, так и для расширения файла;
- маске **\*.mp3** соответствуют все аудиозаписи формата mp3;
- маска **info.\*** позволит отобразить файлы всех типов с именем info;
- маске **info\*.\*** будут соответствовать все файлы, начинающиеся с info;
- маске **\*info.\*** будут соответствовать все файлы, имена которых заканчиваются на info;
- маска **???.\*** позволит отобразить все файлы с трёхбуквенными именами;
- маске **info??.\*** будут соответствовать все файлы, имена которых состоят из шести символов и начинаются с info;
- маска **info??\*.\*** позволяет найти все файлы, имена которых начинаются с info и содержат шесть и более символов.

## Самое главное

Файл — это поименованная совокупность данных определённого размера, размещаемая на внешних устройствах (носителях информации) и рассматриваемая в процессе обработки как единое целое.

Каталог (папка) — это поименованная совокупность файлов и подкаталогов (вложенных каталогов).

Файловая система — это часть ОС, определяющая способ организации, хранения и именования данных на носителе информации.

Для того чтобы «добраться» до нужного файла в многоуровневой файловой структуре, пользователь может перемещаться по каталогам, образующим путь к файлу. Кроме того, можно воспользоваться поиском по маске имени файла.

## Вопросы и задания

1. Что такое файл? Что такое каталог?
2. Согласны ли вы с тем, что каталог — это файл? Обоснуйте свою точку зрения.
3. Что такое файловая система? Какие задачи она решает?
4. Что такое кластер?
5. На рисунке изображён логический диск, разделённый на кластеры. Размер каждого кластера составляет 16 Кбайт. Используемые кластеры — серые, свободные кластеры — белые.

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30
31	32	33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48	49	50
51	52	53	54	55	56	57	58	59	60
61	62	63	64	65	66	67	68	69	70
71	72	73	74	75	76	77	78	79	80
81	82	83	84	85	86	87	88	89	90
91	92	93	94	95	96	97	98	99	100

На диск требуется записать файлы: файл А размером 20 Кбайт, затем файл В размером 30 Кбайт, затем файл С размером 65 Кбайт, и наконец — файл D размером 60 Кбайт. Операционная система для хранения каждого файла пытается искать смежные (соседние) свободные кластеры, чтобы избежать фрагментации файла — рассеивания содержимого файла по диску, приводящего к уменьшению производительности.

В какие кластеры может быть записано содержимое файлов при соблюдении этого условия?



6. В Windows существует три возможных варианта файловой системы: NTFS, FAT32 и редко используемая устаревшая система FAT (также известная как FAT 16). Найдите в Интернете информацию об этих файловых системах. Какая из них является предпочтительной файловой системой для Windows 8?

7. Перемещаясь из одного каталога в другой, пользователь последовательно посетил каталоги Байкал, Путешествия, Фото, Е:, Документы, География, Карты, География, Изображения. При каждом перемещении пользователь либо спускался в каталог на уровень ниже, либо поднимался на уровень выше. В каталоге, из

которого были начаты перемещения, пользователь скопировал файл 1245.jpg. Этот файл он вставил в каталог, в котором оказался в результате своих перемещений, переименовав его в Листвянка.jpg. Укажите полные имена файлов 1245.jpg и Листвянка.jpg.

8. Определите, какое из следующих имён файлов удовлетворяет маске ?ba\*r.?xt.

1) bar.txt; 2) obar.txt; 3) obar.xt; 4) barr.txt.



9. В каталоге находится 6 файлов:

chifera.dat;	deLafer.doc;
chifera.doc;	oferta.doc;
ferrum.doc;	tokoferol.doc.

Определите, с помощью какой маски из каталога будет отобрано ровно три файла.

1) \*fer?\*.d\*; 2) ?fer\*.doc; 3) \*?fer\*?.do\*; 4) \*fer?.doc.