

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА



Институт радиоэлектроники и информационных технологий
Кафедра вычислительные системы и технологии

Лабораторная работа № 3
Файловая система FAT

ОТЧЕТ

по лабораторной работе

по дисциплине

Принципы и методы
организации системных программных средств

РУКОВОДИТЕЛЬ:

_____ Викулова Е.Н.

СТУДЕНТ:

_____ Сапожников В.О.

19-ИВТ-3

Работа защищена «___» _____

С оценкой _____

Нижний Новгород 2022

План работы

Подготовительные действия:

Самостоятельно выбрать *hex*-редактор для работы с файловой системой FAT (*exFAT*), ознакомиться с его функциональными возможностями.

- подготовить носитель с файловой системой *FAT* (USB-флэш-накопитель, внешний диск), или создать FAT раздел на диске.

Используя выбранный *hex*-редактор, уметь выполнять следующие действия:

- работать с диском на физическом (*c,h,s*) и логическом (*LBA*) уровнях;
- находить и читать:
 - а) *данные загрузочной области*:
 - команду перехода к коду загрузки;
 - идентификатор системы,
 - дескриптор носителя;
 - количество байт в секторе;
 - количество секторов в кластере;
 - количество секторов, зарезервированных для загрузочной записи;
 - количество таблиц *FAT*, размер *FAT*;
 - количество записей в корневом каталоге;
 - общее количество секторов на диске;
 - байты сигнатуры загрузочного сектора;
 - и др. (например, при работе с *FAT32*, данные актуальные только для *FAT32*).
 - б) *область FAT*;
 - в) *данные корневого каталога*:
 - имя файла или подкаталога,
 - расширение имени,
 - атрибуты файла,
 - время и дату создания файла,
 - размер файла;
 - г) *данные подкаталогов*;
 - д) *содержимое файлов*:
 - изменять имя, размер и атрибуты файла;
 - создавать, удалять и восстанавливать файлы;
 - объединять файлы;
 - удалять кластеры из файла.

1. Изучить свойство фрагментации дисковой памяти. Выполняя файловые операции добавления, удаления, модификации файлов создать фрагментацию отдельных файлов.

С помощью выбранного *hex*-редактора:

- найти цепочки кластеров, принадлежащих фрагментированным файлам;
- отслеживать изменения при создании новых файлов (каталогов) и модификации файлов (например, при изменении размера файла);

- выявить стратегию выделения дискового пространства, которую использует операционная система.

2. Изучить причины возникновения ошибок логической структуры диска и методы их исправления.

С помощью выбранного *hex*-редактора ввести логические некорректности в файловую структуру (потерянные кластеры, пересекающиеся файлы).

Используя утилиты проверки и восстановления (найти и выбрать самостоятельно), попытаться восстановить файловую структуру диска.

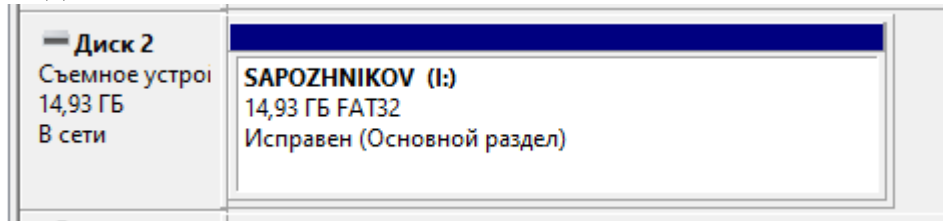
Выбор редактора

WinHex редактор от X-Ways Software Technology AG, распространяемый для ОС Windows. Как и все другие HEX-редакторы, WinHex может отображать контрольные суммы или коды программных файлов, которые не под силу обычным текстовым редакторам.

Ко всему прочему, утилита может использоваться для повседневной работы в аварийных ситуациях для проверки и редактирования всех известных и неизвестных видов файлов, восстановления случайно удаленных файлов из Корзины Windows (и не только) или цифровых карт камеры и многое другое.

Поскольку для данной работы необходим Hex редактор с полной функциональностью, а условная бесплатная версия WinHex такой не обладает, то была использована “пиратская” версия, что мы, конечно же, осуждаем.

Для работы была использован Flash накопитель на 16ГБ, отформатированный в FAT32 при помощи встроенной в Windows утилиты управления дисками.



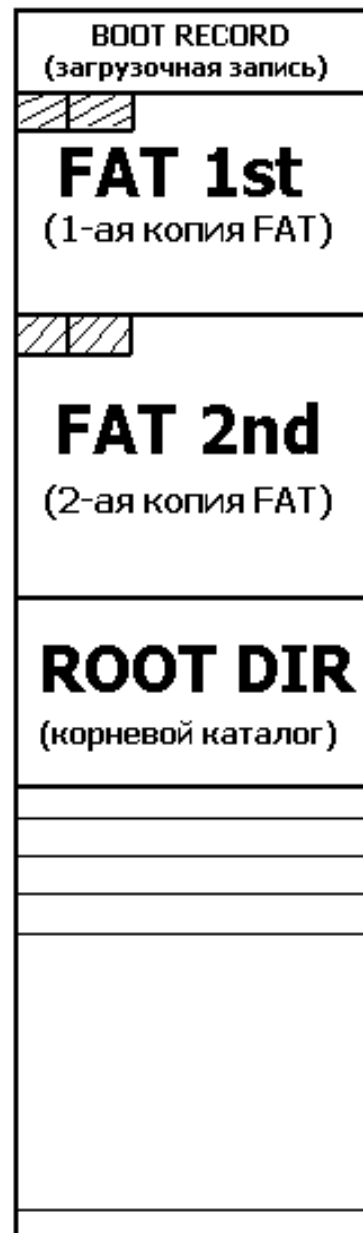
Логическая структура диска FAT

При форматировании диска создается набор системных данных, необходимых для работы с файлами на логическом уровне. Подготовленный носитель файловой системы принято также называть томом (volume).

Логическая структура диска (тома) *FAT* состоит из четырех областей, размеченных статически:

- загрузочная запись (*Boot Record*),
- область таблиц *FAT* (2 копии),
- корневой каталог (*Root Dir*),
- область данных (*Data Area*).

(Root directory)		8,0 KB				32 768
System Volume Information		8,0 KB	02.04.2022 13:33:42	02.04.2022 13:33:44	SH	32 784
Boot sector		1,1 MB				0
FAT 1		7,5 MB				2 224
FAT 2		7,5 MB				17 496
Free space (net)		14,9 GB				
Idle space		0 B				
Volume slack		0,5 KB				31 308 416



Загрузочный сектор *Boot Record*

Первые 3 байта занимает команда перехода к коду загрузки, которая обычно EB5890h (наш случай не стал исключением)

Offset	0	1	2
00000000	EB	58	90

Следующие 8 байт занимает идентификатор OEM ID, указывает какая система форматировала раздел.

3	4	5	6	7	8	9	A
4D	53	44	4F	53	35	2E	30

Следующие 2 байта указывают количество байт в секторе. Обычно 512

B	C
00	02

 Действительно, 200h = 512d

Затем 1 байт занимает кол-во секторов в кластере. Данное число должно быть равно степеням двойки (1, 2, 4, 8, 16, 32, 64). В нашем случае оно равно 10h = 16d

D
10

Следующие 2 байта занимает значение кол-ва секторов, зарезервированных для загрузочной записи(записей).

E	F
B0	08

1 байт на кол-во структур в FAT разделе, обычно 2 (как и в нашем случае)

Offset	
000000010	02

2 байта – кол-во 32-байтовых записей каталога в корневом каталоге разделов FAT12 и FAT16. В разделах FAT32 должно быть равно 0 (так и есть)

1	2
00	00

2байта – число, указывающее общее кол-во секторов в разделах, содержащих менее 65536 секторов. FAT32 должно быть равно 0.

3	4
00	00 1

Для FAT 32 данное значение указано в поле по смещению 20h и занимает 4 байта

000000020	81 BA DD 01
-----------	-------------

1 байт-дескриптор носителя. Для всех несменных носителей обычно F8h (как в нашем случае), для большинства сменных F0h

5
F8

2 байта – число, указывающее общее кол-во секторов, занятых разделами FAT12, FAT16. В FAT32 данной число равно 0.

6	7
00	00

Количество секторов, занятых разделами FAT указано по смещение 24h и занимает 4 байта

4	5	6	7
A8	3B	00	00

Нулевой сектор загрузочной записи раздела FAT32 завершается байтами сигнатуры, значение которых должно быть равно 55Aah.

E	F
55	AA

Дополнительно:

2 байта – кол-во секторов на дорожке для прерывания 13h, на жестких дисках обычно 63.

8	9
3F	00

3Fh = 63d

Следующие 2 байта – число головок для прерывания 13h. На жестких дисках обычно 255

A	B
FF	00

Действительно, FFh = 255d

По смещение 28h, находится информация характерная только для FAT32. Биты 0-3. Номер активного раздела FAT с отсчетом от нуля. Действителен только в том случае, если зеркальное отражение отключено (бит 7 = 1). Биты 4-6. Зарезервированные. Бит 7 – “0” указывает, что раздел FAT зеркально отражен, “1” – что раздел FAT активен. Биты 8-15 зарезервированы.

8	9
00	00

В нашем случае все биты равны нулям, раздел FAT отражен.

Разделы FAT

WinHex обладает удобным меню навигации по диску, которое позволяет сместиться на разделы FAT

Name▲	Ext.	Size	Created	Modified	Recc Attr.	1st sector
(Root directory)		8,0 KB				32 768
System Volume Information		8,0 KB	02.04.2022 13:33:42	02.04.2022 13:33:44	SH	32 784
ИВ.txt	txt	0 B	02.04.2022 14:44:39	02.04.2022 14:44:40	A	
Новый текстовый документ...	txt	0 B	02.04.2022 14:44:39	02.04.2022 14:44:40	A	7 372 768
Boot sector		1,1 MB				0
FAT 1		7,5 MB				2 224
FAT 2		7,5 MB				17 496
Free space (net)		14,9 GB				
Idle space		?				
Volume slack		0,5 KB				31 308 416

FAT1:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII
000116000	F8	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	0F		YYY	YYYYYYYY YYY
000116010	FF	FF	FF	0F	FF	FF	FF	0F	00	00	00	00	00	00	00	00		YYY	YYY
000116020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
000116030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
000116040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
000116050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			

FAT2:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII
00088B000	F8	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	0F		YYY	YYYYYYYY YYY
00088B010	FF	FF	FF	0F	FF	FF	FF	0F	00	00	00	00	00	00	00	00		YYY	YYY
00088B020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
00088B030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
00088B040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
00088B050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
00088B060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			

Теперь посчитаем смещение вручную. Кол-во зарезервированных секторов равно $8B0 \left(\overset{EF}{B0 \ 08} \right)h = 2224d$. Тогда FAT1 находится по смещению $2224 * 512 = 1138688 = 11600h$, куда нас и перенаправил WinHex.

Аналогично для FAT2: $17496 * 512 = 8957952d = 88B000h$

Корневой каталог

Корневой каталог находит по смещению: $1138688 + 15272 * 512 * 2 = 16777216 = 1000000h$

Name	Ext.	Size	Created	Modified	ReccAttr.	1st sector
(Root directory)		8,0 KB				32 768
System Volume Information		8,0 KB	02.04.2022 16:19:08	02.04.2022 16:19:10	SH	32 784
Boot sector		1,1 MB				0
FAT 1		7,5 MB				2 224
FAT 2		7,5 MB				17 496
Free space (net)		14,9 GB				
Idle space		?				
Volume slack		0,5 KB				31 308 416

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII
001000000	53	41	50	4F	5A	48	4E	49	4B	4F	56	08	00	00	00	00		SAPOZHNIKOV
001000010	00	00	00	00	00	00	65	82	82	54	00	00	00	00	00	00		e,,T
001000020	42	20	00	49	00	6E	00	66	00	6F	00	0F	00	72	72	00		B I n f o r r
001000030	6D	00	61	00	74	00	69	00	6F	00	00	00	6E	00	00	00		m a t i o n
001000040	01	53	00	79	00	73	00	74	00	65	00	0F	00	72	6D	00		S y s t e r m
001000050	20	00	56	00	6F	00	6C	00	75	00	00	00	6D	00	65	00		V o l u m e
001000060	53	59	53	54	45	4D	7E	31	20	20	20	16	00	05	64	82		SYSTEM~1 d,
001000070	82	54	82	54	00	00	65	82	82	54	03	00	00	00	00	00		,T,T e,,T
001000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		

Создадим файл Hello.txt с текстом Hello. В корневом каталоге добавилась запись:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII
001000000	53	41	50	4F	5A	48	4E	49	4B	4F	56	08	00	00	00	00		SAPOZHNIKOV
001000010	00	00	00	00	00	00	65	82	82	54	00	00	00	00	00	00		e,,T
001000020	42	20	00	49	00	6E	00	66	00	6F	00	0F	00	72	72	00		B I n f o r r
001000030	6D	00	61	00	74	00	69	00	6F	00	00	00	6E	00	00	00		m a t i o n
001000040	01	53	00	79	00	73	00	74	00	65	00	0F	00	72	6D	00		S y s t e r m
001000050	20	00	56	00	6F	00	6C	00	75	00	00	00	6D	00	65	00		V o l u m e
001000060	53	59	53	54	45	4D	7E	31	20	20	20	16	00	05	64	82		SYSTEM~1 d,
001000070	82	54	82	54	00	00	65	82	82	54	03	00	00	00	00	00		,T,T e,,T
001000080	E5	78	00	74	00	00	00	FF	FF	FF	FF	0F	00	15	FF	FF		âx t ŷŷŷŷ ŷŷ
001000090	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF		ŷŷŷŷŷŷŷŷŷŷ ŷŷŷŷ
0010000A0	E5	4B	04	39	04	20	00	34	04	3E	04	0F	00	15	3A	04		âK 9 4 > :
0010000B0	43	04	3C	04	35	04	3D	04	42	04	00	00	2E	00	74	00		C < 5 = B . t
0010000C0	E5	1D	04	3E	04	32	04	4B	04	39	04	0F	00	15	20	00		â > 2 K 9
0010000D0	42	04	35	04	3A	04	41	04	42	04	00	00	3E	04	32	04		B 5 : A B > 2
0010000E0	E5	8E	82	9B	89	92	7E	31	54	58	54	20	00	0A	8E	84		âŽ, > * ' ~ 1 T X T Ž,,
0010000F0	82	54	82	54	00	00	8F	84	82	54	00	00	00	00	00	00		,T,T ,,T
001000100	41	48	00	65	00	6C	00	6C	00	6F	00	0F	00	F1	2E	00		AH e l l o ñ.
001000110	74	00	78	00	74	00	00	00	FF	FF	00	00	FF	FF	FF	FF		t x t ŷŷ ŷŷŷŷ
001000120	48	45	4C	4C	4F	20	20	20	54	58	54	20	00	0A	8E	84		HELLO TXT Ž,,
001000130	82	54	82	54	00	00	0C	85	82	54	05	00	05	00	00	00		,T,T ...,T
001000140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		

Так же появилась запись и в таблице FAT:

Boot sector																		
FAT 1																		2 224
FAT 2																		17 496

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII
000116000	FF	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	FF		ŷŷŷŷ ŷŷŷŷŷŷŷŷ ŷŷŷŷ
000116010	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	00	00	00	00		ŷŷŷŷ ŷŷŷŷ ŷŷŷŷ

Смещение	Значение	Расшифровка	Описание
000	48454C4C4A202020	“HELLO ”	Имя файла
008	545854	“TXT”	Расширение файла
00B	20	Архивный	Атрибуты файла

00C	Зарезервировано		
1A	0500	5	Начальный кластер
1C	05	5 байт	Размер файла в байтах

Значения 5ого кластера:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII
001006000	48	65	6C	6C	6F	00	00	00	00	00	00	00	00	00	00	00		Hello

Добавление подкаталогов и файлов.

Изменение FАТ:

FAT1		7,5 MB				2 224
FAT2		7,5 MB				17 496

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII
000116000	F8	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	0F	øÿÿ	ÿÿÿÿÿÿÿÿ	ÿÿÿ
000116010	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	ÿÿÿ	ÿÿÿ	ÿÿÿ
000116020	FF	FF	FF	0F	00	00	00	00	00	00	00	00	00	00	00	00	ÿÿÿ		
000116030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			

[illegible]

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII
00100A000	2E	20	20	20	20	20	20	20	20	20	20	10	00	87	AE	A9	.		#00
00100A010	82	54	82	54	00	00	AF	A9	82	54	07	00	00	00	00	00	,T,T	©,T	
00100A020	2E	2E	20	20	20	20	20	20	20	20	20	10	00	87	AE	A9	..		#00
00100A030	82	54	82	54	00	00	AF	A9	82	54	00	00	00	00	00	00	,T,T	©,T	
00100A040	E5	78	00	74	00	00	00	FF	FF	FF	FF	0F	00	15	FF	FF	âx t	yyyy	yy
00100A050	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy	yyyy	
00100A060	E5	4B	04	39	04	20	00	34	04	3E	04	0F	00	15	3A	04	âK 9	4 >	:
00100A070	43	04	3C	04	35	04	3D	04	42	04	00	00	2E	00	74	00	C < 5 = B	. t	
00100A080	E5	1D	04	3E	04	32	04	4B	04	39	04	0F	00	15	20	00	â > 2 K 9		
00100A090	42	04	35	04	3A	04	41	04	42	04	00	00	3E	04	32	04	B 5 : A B	> 2	
00100A0A0	E5	8E	82	9B	89	92	7E	31	54	58	54	20	00	7D	B2	A9	âŽ, %' ~1TXT	} ©	
00100A0B0	82	54	82	54	00	00	B3	A9	82	54	00	00	00	00	00	00	,T,T	©,T	
00100A0C0	4A	41	43	4B	20	20	20	20	54	58	54	20	18	7D	B2	A9	JACK	TXT	} ©
00100A0D0	82	54	82	54	00	00	C2	A9	82	54	08	00	A6	00	00	00	,T,T	â€,T	;

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII
00100A0D0	82	54	82	54	00	00	C2	A9	82	54	08	00	A6	00	00	00		,T,T	Ä,T

[illegible]

Изменим содержимое файла, удалив 2 последние строки.

Изменился размер файла и дата (Кластер 7, подкаталог Sapozhnikov)

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII
00100A000	2E	20	20	20	20	20	20	20	20	20	20	10	00	87	AE	A9	.	+@
00100A010	82	54	82	54	00	00	AF	A9	82	54	07	00	00	00	00	00	,T,T	^@,T
00100A020	2E	2E	20	20	20	20	20	20	20	20	20	10	00	87	AE	A9	..	+@
00100A030	82	54	82	54	00	00	AF	A9	82	54	00	00	00	00	00	00	,T,T	^@,T
00100A040	E5	78	00	74	00	00	00	FF	FF	FF	FF	0F	00	15	FF	FF	ax t	Y Y Y Y Y Y
00100A050	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	Y Y Y Y Y Y Y Y Y Y	Y Y Y Y Y
00100A060	E5	4B	04	39	04	20	00	34	04	3E	04	0F	00	15	3A	04	AK 9	4 > :
00100A070	43	04	3C	04	35	04	3D	04	42	04	00	00	2E	00	74	00	C < 5 = B	. t
00100A080	E5	1D	04	3E	04	32	04	4B	04	39	04	0F	00	15	20	00	A > 2 K 9	
00100A090	42	04	35	04	3A	04	41	04	42	04	00	00	3E	04	32	04	B 5 : A B	> 2
00100A0A0	E5	8E	82	9B	89	92	7E	31	54	58	54	20	00	7D	B2	A9	az, > % ' ~ 1 TXT	} ^
00100A0B0	82	54	82	54	00	00	B3	A9	82	54	00	00	00	00	00	00	,T,T	^@,T
00100A0C0	4A	41	43	4B	20	20	20	20	54	58	54	20	18	7D	B2	A9	JACK	TXT } ^
00100A0D0	82	54	82	54	00	00	D6	AA	82	54	08	00	5B	00	00	00	,T,T	Ö^,T [

И так же изменилось само содержимое файла.

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII
00100C000	54	68	69	73	20	69	73	20	74	68	65	20	68	6F	75	73	This is the hous	
00100C010	65	20	74	68	61	74	20	4A	61	63	6B	20	62	75	69	6C	e that Jack buil	
00100C020	74	2E	0D	0A	54	68	69	73	20	69	73	20	74	68	65	20	t. This is the	
00100C030	6D	61	6C	74	20	74	68	61	74	20	6C	61	79	20	69	6E	malt that lay in	
00100C040	20	74	68	65	20	68	6F	75	73	65	20	74	68	61	74	20	the house that	
00100C050	4A	61	63	6B	20	62	75	69	6C	74	2E	00	00	00	00	00	Jack built.	

Теперь удалим текстовый файл стандартным способом, не через WinHex

Изменилась цепочка кластеров в FAT

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII
000116000	F8	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	FF	øY Y Y Y Y Y Y Y Y Y	
000116010	FF	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	FF	Y Y Y Y Y Y Y Y Y Y	
000116020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		

В начале имени файла добавился байт состояния – E5, т.е. файл удален

00100A0B0	82	54	82	54	00	00	B3	A9	82	54	00	00	00	00	00	00	,T,T	^@,T
00100A0C0	E5	41	43	4B	20	20	20	20	54	58	54	20	18	7D	B2	A9	ACK	TXT } ^
00100A0D0	82	54	82	54	00	00	D6	AA	82	54	08	00	5B	00	00	00	,T,T	Ö^,T [

При этом сами данные (8 кластер) остались:

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII
00100C000	54	68	69	73	20	69	73	20	74	68	65	20	68	6F	75	73	This is the hous	
00100C010	65	20	74	68	61	74	20	4A	61	63	6B	20	62	75	69	6C	e that Jack buil	
00100C020	74	2E	0D	0A	54	68	69	73	20	69	73	20	74	68	65	20	t. This is the	
00100C030	6D	61	6C	74	20	74	68	61	74	20	6C	61	79	20	69	6E	malt that lay in	
00100C040	20	74	68	65	20	68	6F	75	73	65	20	74	68	61	74	20	the house that	
00100C050	4A	61	63	6B	20	62	75	69	6C	74	2E	00	00	00	00	00	Jack built.	

Повреждение и восстановление файлов.

Дополним файл до 18.6Кб, чтобы он занимал несколько кластеров. Число занятый кластеров в таблице FAT изменилось, добавилось еще 3 кластера.

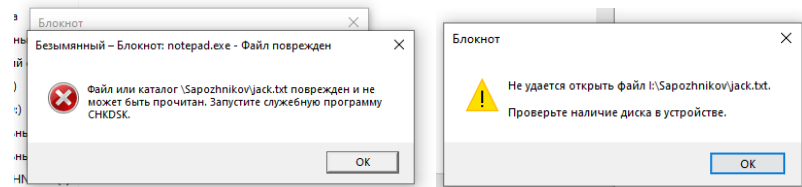
(Root directory)	8,0 KB					32 768
System Volume Information	8,0 KB	02.04.2022 16:19:08	02.04.2022 16:19:10		SH	32 784
Boot sector	1,1 MB					0
FAT 1	7,5 MB					2 224
FAT 2	7,5 MB					17 496
Free space (net)	14,9 GB					
Idle space	?					
Volume slack	0,5 KB					31 308 416

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII
000116000	F8	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	0F	øÿÿ	ÿÿÿÿÿÿÿÿ	ÿÿÿ
000116010	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	ÿÿÿ	ÿÿÿ	ÿÿÿ
000116020	09	00	00	00	0A	00	00	00	FF	FF	FF	0F	00	00	00	00		ÿÿÿ	
000116030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			

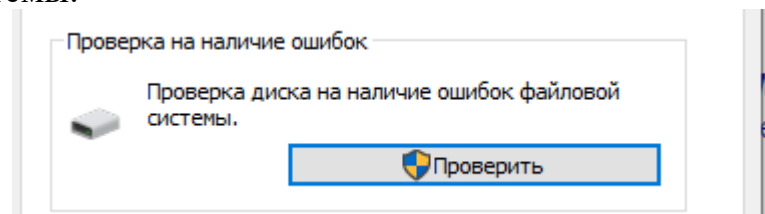
3. Стираем последний кластер в цепочке. Размер цепочки станет 2 вместо

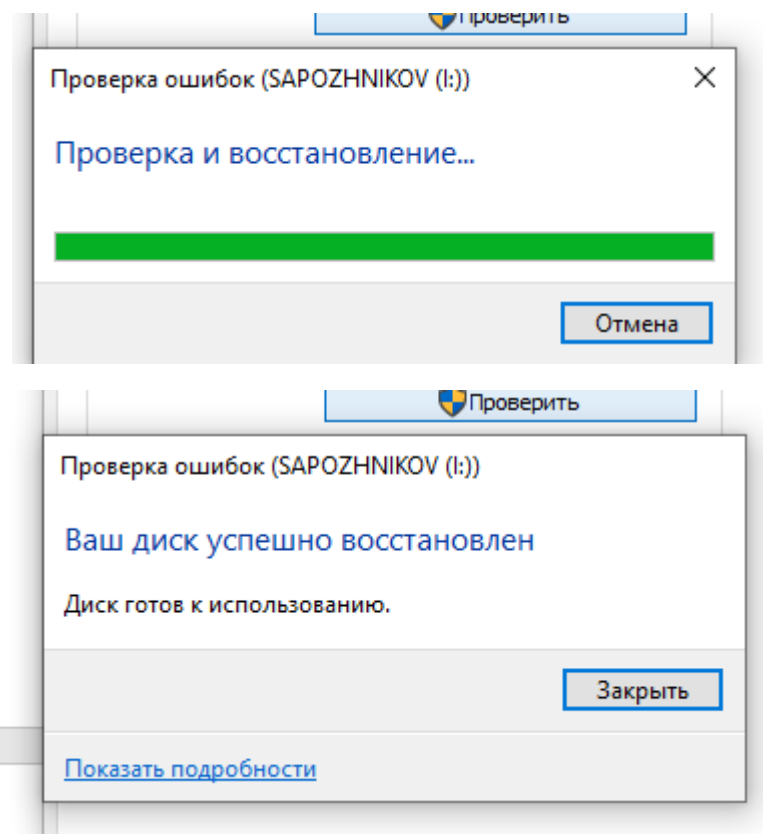
[illegible]

Теперь данный файл распознается как поврежденный

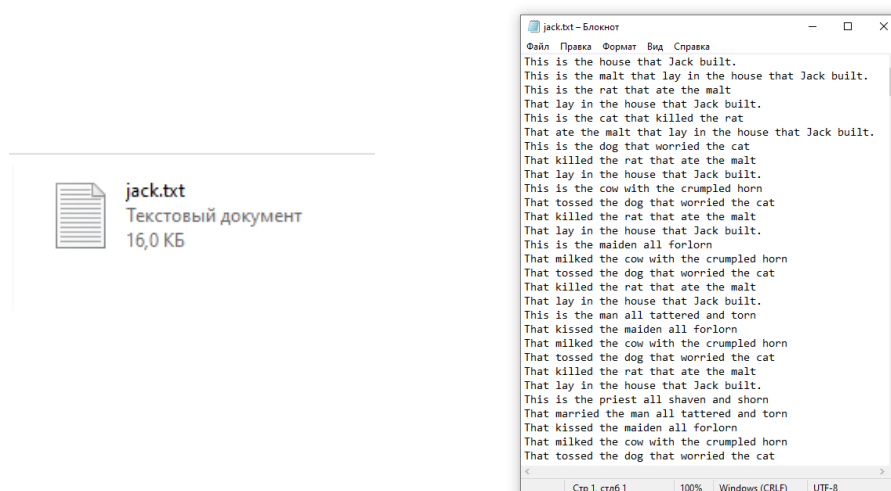


Запустим встроенную в Windows утилиту поиска и справления ошибок файловой системы.





Диск был восстановлен, файл открывается, но его размер уменьшился. Файл был урезан ровно до того, где мы урезали кластер.

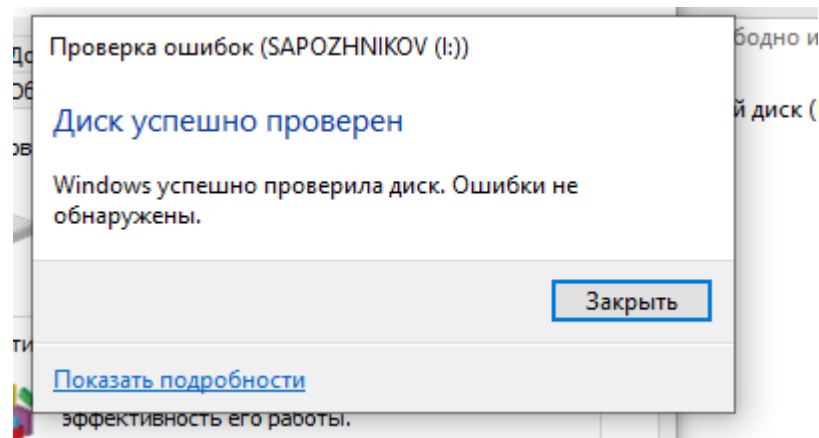
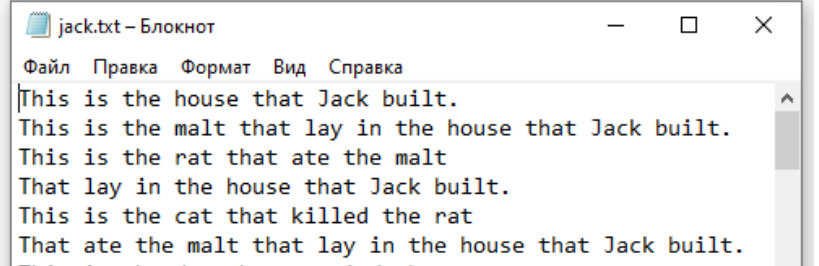
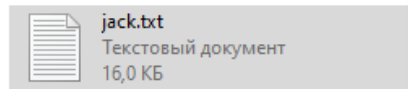


“Хвост” файла был попросту отрезан.

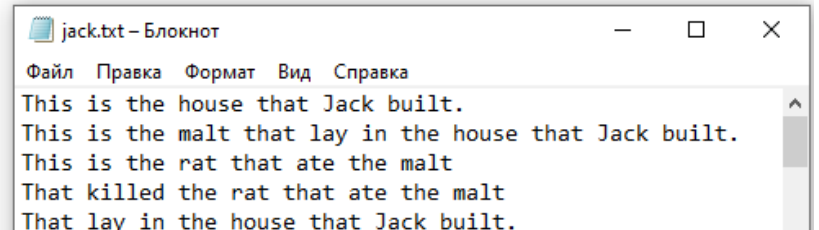
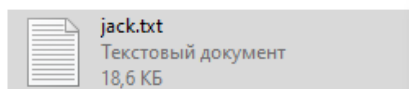
Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII
000116000	F8	FF	FF	0F	FF	FF	FF	7F	FF	FF	FF	0F	FF	FF	FF	0F	øÿÿ	ÿÿÿ	ÿÿÿ
000116010	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	ÿÿÿ	ÿÿÿ	ÿÿÿ
000116020	09	00	00	00	FF	FF	FF	0F	00	00	00	00	00	00	00	00	ÿÿÿ		
000116030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			

Попытаемся вернуть прежний размер файла восстановив последовательность разделов.

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII	
000116000	F8	FF	FF	0F	FF	FF	FF	7F	FF	FF	FF	0F	FF	FF	FF	0F	øÿÿ	ÿÿÿ	ÿÿÿ	ÿÿÿ
000116010	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	ÿÿÿ	ÿÿÿ	ÿÿÿ	ÿÿÿ
000116020	09	00	00	00	0A	00	00	00	FF	FF	FF	0F	00	00	00	00		ÿÿÿ		



К очень большому удивлению, размер файла и его данные полностью восстановились и нам не пришлось вручную переписывать размер файла.



Фрагментация файлов

Устроим фрагментацию. Сначала для удобства отформатируем диск

[illegible]

Добавим 2 файла: main.txt и temp.txt

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI ASCII	
00088B000	F8	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	0F	øÿÿ	ÿÿÿÿÿÿÿ	ÿÿÿ
00088B010	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	FF	FF	FF	0F	ÿÿÿ	ÿÿÿ	ÿÿÿ

А теперь допишем в temp.txt больше данных.

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	V	ANSI	ASCII
00088B000	F8	FF	FF	0F	FF	FF	FF	FF	FF	FF	FF	0F	FF	FF	FF	0F	øÿÿ	ÿÿÿÿÿÿÿ	ÿÿÿ
00088B010	FF	FF	FF	0F	FF	FF	FF	0F	08	00	00	00	FF	FF	FF	0F	ÿÿÿ	ÿÿÿ	ÿÿÿ
00088B020	09	00	00	00	0A	00	00	00	0B	00	00	00	0C	00	00	00			
00088B030	0D	00	00	00	FF	FF	FF	0F	00	00	00	00	00	00	00	00	ÿÿÿ		

Красный – temp.txt

Синий – main.txt

Сама фрагментация на границах каталогов. Видно, что текст песни группы Rammstein – Du Hast прерывается стихотворением про Дом, который построил Джек.

001009FB0	74 0D 0A 64 75 20 68 61	73 74 20 6D 69 63 68 20	t du hast mich
001009FC0	5B 32 78 5D 0D 0A 0D 0A	44 75 0D 0A 64 75 20 68	[2x] Du du h
001009FD0	61 73 74 0D 0A 64 75 20	68 61 73 74 20 6D 69 63	ast du hast mic
001009FE0	68 20 28 32 78 29 0D 0A	64 75 20 68 61 73 74 20	h (2x) du hast
001009FF0	6D 69 63 68 20 67 65 66	72 61 67 74 20 28 32 78	mich gefragt (2x
00100A000	54 68 69 73 20 69 73 20	74 68 65 20 68 6F 75 73	This is the hous
00100A010	65 20 74 68 61 74 20 4A	61 63 6B 20 62 75 69 6C	e that Jack buil
00100A020	74 2E 0D 0A 54 68 69 73	20 69 73 20 74 68 65 20	t. This is the
00100A030	6D 61 6C 74 20 74 68 61	74 20 6C 61 79 20 69 6E	malt that lay in
00100A040	20 74 68 65 20 68 6F 75	73 65 20 74 68 61 74 20	the house that
00100BFC0	20 4A 61 63 6B 20 62 75	69 6C 74 2E 0D 0A 54 68	Jack built. Th
00100BFD0	69 73 20 69 73 20 74 68	65 20 70 72 69 65 73 74	is is the priest
00100BFEO	20 61 6C 6C 20 73 68 61	76 65 6E 20 61 6E 64 20	all shaven and
00100BFF0	73 68 6F 72 6E 0D 0A 54	68 61 74 20 6D 61 72 72	shorn That marr
00100C000	29 0D 0A 64 75 20 68 61	73 74 20 6D 69 63 68 20) du hast mich
00100C010	67 65 66 72 61 67 74 2C	20 75 6E 64 20 69 63 68	gefragt, und ich
00100C020	20 68 61 62 20 6E 69 63	68 74 73 20 67 65 73 61	hab nichts gesa
00100C030	67 74 0D 0A 0D 0A 57 69	6C 6C 73 74 20 64 75 20	gt Willst du
00100C040	62 69 73 20 64 65 72 20	54 6F 64 20 65 75 63 68	bis der Tod euch

Дефрагментация.

Твердотельные накопители и флешки на основе различных видов флеш-памяти в дефрагментации файловых систем не нуждаются. Более того, дефрагментация, в некоторой степени, вредит флеш-памяти, так как последняя имеет намного более ограниченное количество циклов записи/перезаписи, чем накопители на магнитных дисках при должном использовании.

Сама идея дефрагментации состоит в перераспределении фрагментов файлов и логических структур файловых систем на дисках для обеспечения последовательности кластеров.

Вывод

Файловая система FAT32 хорошо подходит для различных носителей. Она имеет высокую скорость работы и достаточно эффективна при работе с файлами небольшого размера. Также она обеспечивает более низкий износ дисков, вследствие меньшего количества передвижений головок чтения/записи. Однако она не ведёт журналирования и поэтому имеет низкую защиту от сбоев.

Для работы с данными в исходном виде достаточно простого Нех-редактора, который позволяет просматривать данные в hex и ascii форматах. WinHex имеет дополнительные преимущества в виде меню навигации по секторам и разделам диска, однако данное ПО является платным, а использование “пиратских” версий мы осуждаем.