

ОГЛАВЛЕНИЕ

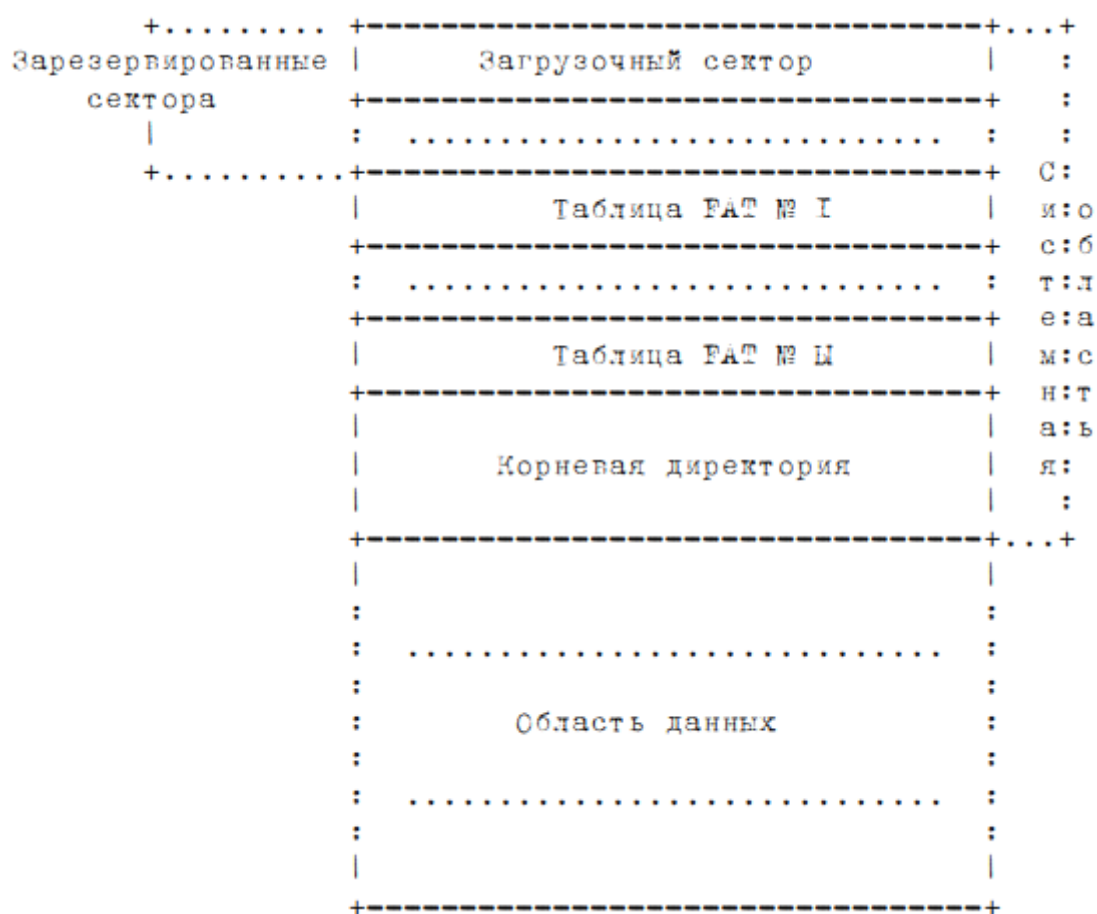
ОСНОВНАЯ ИНФОРМАЦИЯ.....	2
ФАЙЛОВАЯ СИСТЕМА FAT.....	3
Шаг 0. Конфуций сказал.....	3
Шаг 1. Загрузочный сектор.....	3
Шаг 2. Корневая директория.....	3
Шаг 3. Область данных.....	3
Шаг 4. Директория.....	4
Шаг 5. Файл.....	4
Шаг 6. Кластеры файла.....	4
Шаг 7. Выгрузка кластеров файла.....	6
Шаг 8. Подготовка секторов.....	6
Шаг 9. Объединение секторов.....	6
Шаг 10. Обрезание размера файла.....	6
Шаг 11. Вычисление контрольной суммы.....	6
ПРИМЕР РЕШЕНИЯ ЗАДАЧИ НА FAT12.....	7
ФАЙЛОВАЯ СИСТЕМА NTFS.....	16

ОСНОВНАЯ ИНФОРМАЦИЯ

1) Внутри зон байты читаются справа налево, но сами байты слева направо.

EB F0 32 E4 CD
0xCDE432F0EB

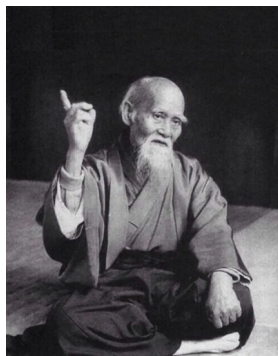
2) Структура FAT:



3) АБОБА

ФАЙЛОВАЯ СИСТЕМА FAT

Шаг 0. Конфуций сказал



"Когда три знака являются взору твоему — F, да A, да T —
не спеши роптать и сетовать на судьбу.
Ибо для одних это испытание,
для других — милость небес.
Мудрый радуется удаче,
глупец же винит FAT в содеянном.
Помни: даже великий диск некогда был пуст."

Конфуций

Шаг 1. Загрузочный сектор

```
offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000 EB 3C 90 6D 6B 66 73 2E 66 61 74 00 02 02 06 00  ě<mkfs.fat....
00000010 03 00 02 00 20 F8 0C 00 20 00 02 00 00 00 00 00  ....ø..
00000020 00 00 00 00 80 00 29 C4 83 4A 39 4E 4F 20 4E 41| ....B.)ÄJ9NO NA
00000030 4D 45 20 20 20 20 46 41 54 31 32 20 20 20 0E 1F| ME FAT12 ..
00000040 BE 5B 7C AC 22 C0 74 0B 56 B4 0E BB 07 00 CD 10| ж[-"Ät.V'.»..í.
00000050 5E EB F0 32 E4 CD 16 CD 19 EB FE 54 68 69 73 20| ^ëð2äí.í.ëþThis
00000060 69 73 20 6E 6F 74 20 61 20 62 6F 6F 74 61 62 6C| is not a bootabl
00000070 65 20 64 69 73 6B 2E 20 20 50 6C 65 61 73 65 20| e disk. Please
00000080 69 6E 73 65 72 74 20 61 20 62 6F 6F 74 61 62 6C| insert a bootabl
00000090 65 20 66 6C 6F 70 70 79 20 61 6E 64 0D 0A 70 72| e floppy and..pr
000000A0 65 73 73 20 61 6E 79 20 6B 65 79 20 74 6F 20 74| ess any key to t
000000B0 72 79 20 61 67 61 69 6E 20 2E 2E 2E 20 0D 0A 00| ry again ... ..
```

Система хранения

Размер сектора в байтах

Размер кластера в секторах

Количество зарезервированных секторов

Количество FAT-таблиц

Размер корневой директории в байтах

Размер FAT-таблицы

Шаг 2. Корневая директория

Формула для нахождения первого сектора корневой директории:

Количество зарезервированных секторов + *Количество FAT-таблиц* · *Размер FAT-таблицы*.

Шаг 3. Область данных

Формула для нахождения размера корневой директории в секторах:

$$\frac{\text{Размер корневой директории в байтах} \cdot \text{Количество строк в секторе}}{\text{Размер сектора в байтах}},$$

где *Количество строк в секторе* = 32.

Формула для нахождения начала области данных:

Первый сектор корневой директории + *Размер корневой директории в секторах*.

Шаг 4. Директория

НАЧАЛО ПРОЦЕССА

offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00005400	E5	55	4E	4B	20	20	20	20	20	20	20	10	08	00	60	74	åUNK ...`t
00005410	BF	5A	BF	5A	00	00	60	74	BF	5A	02	00	00	00	00	00	ěZěZ..`těZ.....
00005420	41	52	00	7A	00	78	00	79	00	58	00	0F	00	68	00	00	AR.z.x.y.X...h..
00005430	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyyyy..yyyy
00005440	52	5A	58	59	58	20	20	20	20	20	20	10	00	00	63	74	RZXYX ...ct
00005450	BF	5A	BF	5A	00	00	63	74	BF	5A	42	00	00	00	00	00	ěZěZ..ctěZB.....

Формула для перемещения к первому кластеру директории:

(Номер первого кластера директории – 2) · Размер кластера в секторах + Начало области данных, так как индексация кластеров начинается с 2.

КОНЕЦ ПРОЦЕССА

Данный процесс повторяем до тех пор, пока не доберёмся до файла.

Шаг 5. Файл

offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F		
00015800	2E	20	20	20	20	20	20	20	20	20	20	10	00	00	63	74ct
00015810	BF	5A	BF	5A	00	00	63	74	BF	5A	33	00	00	00	00	00	ħZħZ...ctħZ3.....	Номер первого кластера файла
00015820	2E	2E	20	20	20	20	20	20	20	20	20	10	00	00	63	74ct
00015830	BF	5A	BF	5A	00	00	63	74	BF	5A	2F	00	00	00	00	00	ħZħZ...ctħZ/.....	Размер файла
00015840	41	79	00	49	00	75	00	7A	00	63	00	0F	00	46	00	00	Ay.I.u.z.c...F..	
00015850	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyyyy..yyyy	
00015860	59	49	55	5A	43	20	20	20	20	20	20	00	00	63	74	YIUZC	..ct	
00015870	BF	5A	BF	5A	00	00	63	74	BF	5A	92	00	F6	10	00	00	ħZħZ...ctħZ0.ö...	

Формула для нахождения размера файла в байтах:

$$0xРазмер\ файла_{16} = Размер\ файла\ в\ байтах_{10}.$$

Формула для нахождения количества кластеров для хранения файла:

$$\left[\frac{\text{Размер файла в байтах}}{\text{Размер сектора в байтах} \cdot \text{Размер кластера в секторах}} \right]$$

Шаг 6. Кластеры файла

Формула для перемещения к первой FAT-таблице:

Количество зарезервированных секторов.

```

offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000C00 F8 FF FF 00 60 01 FF 3F 02 4A C0 00 17 40 01 00| øÿÿ.`.ÿ?.JÀ..@..
00000C10 10 09 35 F0 FF 1E 70 04 3A 00 00 98 80 01 89 10| ..5ðÿ.p:...ðð.ð.
00000C20 03 7B 60 02 B5 00 00 32 F0 FF 48 A0 06 AA D0 04| .{`.м..2ðÿН .æð.
00000C30 00 00 00 41 B0 06 FF EF 0B 51 00 00 FF FF FF 3B| ...A°.ÿİ.Q..ÿÿÿ;
00000C40 D0 07 38 E0 06 00 F0 FF FF CF 03 B6 F0 FF 3F 30| ð.8à..ðÿÿİ.ğðÿ?ð

```

FAT12Кластер занимает **1,5 байта** в FAT-таблице:

```

offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000C00 F8 FF FF 00 60 01 FF 3F 02 4A C0 00 17 40 01 00| øÿÿ.`.ÿ?.JÀ..@..
00000C10 10 09 35 F0 FF 1E 70 04 3A 00 00 98 80 01 89 10| ..5ðÿ.p:...ðð.ð.
00000C20 03 7B 60 02 B5 00 00 32 F0 FF 48 A0 06 AA D0 04| .{`.м..2ðÿН .æð.
00000C30 00 00 00 41 B0 06 FF EF 0B 51 00 00 FF FF FF 3B| ...A°.ÿİ.Q..ÿÿÿ;
00000C40 D0 07 38 E0 06 00 F0 FF FF CF 03 B6 F0 FF 3F 30| ð.8à..ðÿÿİ.ğðÿ?ð

```

FAT16Кластер занимает **2 байта** в FAT-таблице:

```

offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000C00 F8 FF FF 00 60 01 FF 3F 02 4A C0 00 17 40 01 00| øÿÿ.`.ÿ?.JÀ..@..
00000C10 10 09 35 F0 FF 1E 70 04 3A 00 00 98 80 01 89 10| ..5ðÿ.p:...ðð.ð.
00000C20 03 7B 60 02 B5 00 00 32 F0 FF 48 A0 06 AA D0 04| .{`.м..2ðÿН .æð.
00000C30 00 00 00 41 B0 06 FF EF 0B 51 00 00 FF FF FF 3B| ...A°.ÿİ.Q..ÿÿÿ;
00000C40 D0 07 38 E0 06 00 F0 FF FF CF 03 B6 F0 FF 3F 30| ð.8à..ðÿÿİ.ğðÿ?ð

```

НАЧАЛО ПРОЦЕССАФормула для нахождения кластера в FAT-таблице:*Смещение FAT-таблицы + Номер кластера · Размер кластера в FAT-таблице.***FAT12****Целое число**

Переводим в 16-ую систему счисления.

Дробное число

Округляем в меньшую сторону и переводим в 16-ую систему счисления.

FAT16

Переводим в 16-ую систему счисления.

В получившейся 16-ой записи отделяем последнюю цифру. Теперь оставшаяся часть указывает на строку, а цифра – на столбец. На пересечении этих значений будет **байт**, на который указывал кластер.

FAT12**Целое число**Выбираем **байт** и 2 после него:

```

09 35 F0 FF 1E 70 00 09 35 F0 FF 1E 70
B 60 02 B5 00 00 7B 60 02 B5 00 00
A 00 41 B0 06 FF 00 00 00 41 B0 06 FF

```

Меняем местами первый и третий байты. Делим посередине получившийся результат.

Целое число

Берём правую часть

Дробное число

Берём левую часть

FAT16Выбираем **байт** и его соседа справа:

```

35 F0 FF 1E 70
60 02 B5 00 00
00 41 B0 06 FF

```

Меняем местами байты.

Переводим получившейся значение в 10-ую систему счисления.

КОНЕЦ ПРОЦЕССА

Данный процесс надо повторять до тех пор, пока не дойдем до последнего кластера файла. Он должен указывать на FFF в случае FAT12 или FF в случае FAT16.

Шаг 7. Выгрузка кластеров файла

Выгружаем все кластеры. Для этого надо скачать все секторы каждого кластера. Не забываем, что в загрузочном секторе указывался *Размер кластера в секторах*.

Формула для перехода к нужному кластеру:

$(\text{Номер кластера} - 2) \cdot \text{Размер кластера в секторах} + \text{Начало области данных}$.

Если *Размер кластера в секторах* не равен 1, то требуется выгрузить ещё *Размер кластера в секторах* - 1 следующих за ним секторов.

По итогу должно быть выгружено:

$\text{Размер кластера в секторах} \cdot \text{Количество кластеров для хранения файла}$ файлов.

Шаг 8. Подготовка секторов

Переместим все выгруженные секторы в отдельную папку, затем переименуем их в порядке скачивания. Для удобства лучше использовать натуральные числа.

Шаг 9. Объединение секторов

С помощью команды *cat* объединим все секторы в один файл.

Пример: *cat 1 2 3 4 > объединенный файл*.

Затем можно проверить размер полученного файла с помощью команды *ls -l*. Он должен равняться значению:

$\text{Размер сектора в байтах} \cdot \text{Размер кластера в секторах} \cdot \text{Количество кластеров для хранения файла}$.

Шаг 10. Обрезание размера файла

С помощью команды *dd* обрежем размер объединённого файла до *Размера файла в байтах*.

Общий вид команды:

dd if=объединенный файл of=конечный файл bs=1 count=\$((размер файла в байтах)).

Шаг 11. Вычисление контрольной суммы

С помощью команды *md5sum* вычислим контрольную сумму.

Общий вид команды: *md5sum конечный файл*.

Вывод команды копируем без имени файла и вставляем на сайте.

ПРИМЕР РЕШЕНИЯ ЗАДАЧИ НА FAT12

Как только на экране появляется загрузочный сектор, начинаем его анализ. Поле «Для заметок» будем использовать как рабочую тетрадь, куда будем вносить исходные данные и решение:

offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000000	EB	3C	90	6D	68	66	73	2E	66	61	74	00	02	01	05	00
00000010	04	00	02	00	08	F8	06	00	10	00	02	00	00	00	00	00
00000020	00	00	00	00	80	00	29	01	06	5E	1C	4E	4F	20	4E	41
00000030	4D	45	20	20	20	20	46	41	54	31	32	20	20	20	0E	1F
00000040	8E	5B	7C	AC	22	C0	74	0B	56	B4	0E	8B	07	00	CD	10
00000050	5E	EB	F0	32	E4	CD	16	CD	19	EB	FE	54	68	69	73	20
00000060	69	73	20	6E	6F	74	20	61	20	62	6F	6F	74	61	62	6C
00000070	65	20	64	69	73	6B	2E	20	20	50	6C	65	61	73	65	20
00000080	69	6E	73	65	72	74	20	61	20	62	6F	6F	74	61	62	6C
00000090	65	20	66	6C	6F	70	70	79	20	61	6E	64	00	0A	70	72
000000A0	65	73	73	20	61	6E	79	20	68	65	79	20	74	6F	20	74
000000B0	72	79	20	61	67	61	69	6E	20	2E	2E	2E	20	00	0A	00
000000C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
000000F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Имя файла

IMzEu/IMzEu/CHgiu/IUvWe

Для заметок

IMzEu/IMzEu/CHgiu/IUvWe

Система хранения - FAT12
Размер сектора в байтах - 0x0200 = 512 байт
Размер кластера в секторах - 0x01 = 1
Количество зарезервированных секторов - 0x0005 = 5
Количество FAT-таблиц - 0x04 = 4
Размер корневой директории в байтах - 0x0200 = 512 байт
Размер FAT-таблицы - 0x0006 = 6

После внесения имеющихся данных рассчитаем основные значения, которые пригодятся в будущем для нахождения директорий, файла и секторов.

Формула для нахождения первого сектора корневой директории:

Количество зарезервированных секторов + *Количество FAT-таблиц* · *Размер FAT-таблицы*.

Формула для нахождения размера корневой директории в секторах:

$$\frac{\text{Размер корневой директории в байтах} \cdot \text{Количество строк в секторе}}{\text{Размер сектора в байтах}},$$

где *Количество строк в секторе* = 32.

Формула для нахождения начала области данных:

Первый сектор корневой директории + *Размер корневой директории в секторах*.

Внесём в поле «Для заметок» эти результаты:

Первый сектор корневой директории - $5 + 4 \cdot 6 = 29$
Размер корневой директории в секторах - $512 \cdot 32 / 512 = 32$
Начало области данных - $29 + 32 = 61$

Имея эти данные, перемещаемся в корневую директорию. Для этого в поле «Номер сектора» запишем значение Первого сектора корневой директории. Здесь мы замечаем директории и файлы. Находим директорию с интересующим нас именем, то есть «IMzEu». По значению Атрибута проверяем, что это действительно директория (атрибут должен иметь значение 10). Затем опускаем глаза на строку ниже и замечаем

Номер первого кластера директории. Подставим его значение в формулу для перемещения к первому кластеру директории:

$(\text{Номер первого кластера директории} - 2) \cdot \text{Размер кластера в секторах} + \text{Начало области данных}$

Не стесняемся использовать «Калькулятор», ведь сдать зачёт по дисциплине «Операционные системы» очень важно, а для этого надо сначала получить допуск. Так что не рассчитываем на математику в своей голове, а тщательно всё проверяем. Также занесём результаты вычисления в поле «Для заметок». Теперь мы имеем номер сектора для директории «lMzEu».

offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00003A00	E5	55	4E	4B	20	20	20	20	20	20	20	10	08	00	78	85	ÆUNK ...x
00003A10	BF	5A	BF	5A	00	00	78	85	BF	5A	02	00	00	00	00	00	¿Z¿Z...x¿Z¿Z....
00003A20	41	6C	00	4D	00	7A	00	45	00	75	00	0F	00	F1	00	00	A1.M.z.E.u...ñ..
00003A30	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy...yy
00003A40	4C	4D	5A	45	55	20	20	20	20	20	20	10	00	00	7B	85	LMZEU ...{
00003A50	BF	5A	BF	5A	00	00	7B	85	BF	5A	28	00	00	00	00	00	¿Z¿Z...{¿Z¿Z....
00003A60	41	6C	00	55	00	76	00	57	00	65	00	0F	00	15	00	00	A1.U.v.w.e.....
00003A70	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy...yy
00003A80	4C	55	56	57	45	20	20	20	20	20	20	10	00	00	7B	85	LUVWE ...{
00003A90	BF	5A	BF	5A	00	00	7B	85	BF	5A	11	00	00	00	00	00	¿Z¿Z...{¿Z¿Z....
00003AA0	48	51	44	4C	50	20	20	20	20	20	20	20	00	00	7C	85	HQDLP ..{
00003AB0	BF	5A	BF	5A	00	00	7C	85	BF	5A	05	00	89	08	00	00	¿Z¿Z... ¿Z¿Z...
00003AC0	41	43	00	48	00	67	00	69	00	75	00	0F	00	62	00	00	AC.H.g.i.u...b..
00003AD0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy...yy
00003AE0	43	48	47	49	55	20	20	20	20	20	20	20	00	00	7D	85	CHGIU ..}
00003AF0	BF	5A	BF	5A	00	00	7D	85	BF	5A	D1	00	4B	05	00	00	¿Z¿Z...}¿Z¿Z...K...
00003B00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Номер сектора

29

Калькулятор

DEC: 99

HEX: 0x63

Счеты

(0x0028-2)*1+61

Вычислить

lMzEu: (0x0028-2)*1+61=99

Переходим в директорию «lMzEu», введя полученное значение в поле «Номер сектора». Здесь всё то же самое: директории и файлы. Повторяем те же действия, чтоо и раньше и переходим дальше, тоже в директорию с именем «lMzEu»:

offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
0000C600	2E	20	20	20	20	20	20	20	20	20	20	10	00	00	7B	85{
0000C610	BF	5A	BF	5A	00	00	7B	85	BF	5A	28	00	00	00	00	00	¿Z¿Z...{¿Z¿Z....
0000C620	2E	2E	20	20	20	20	20	20	20	20	20	10	00	00	7B	85{
0000C630	BF	5A	BF	5A	00	00	7B	85	BF	5A	00	00	00	00	00	00	¿Z¿Z...{¿Z¿Z....
0000C640	41	6C	00	55	00	76	00	57	00	65	00	0F	00	15	00	00	A1.U.v.w.e.....
0000C650	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy...yy
0000C660	4C	55	56	57	45	20	20	20	20	20	20	10	00	00	7B	85	LUVWE ...{
0000C670	BF	5A	BF	5A	00	00	7B	85	BF	5A	5F	00	00	00	00	00	¿Z¿Z...{¿Z¿Z....
0000C680	48	51	44	4C	50	20	20	20	20	20	20	20	00	00	7B	85	HQDLP ..{
0000C690	BF	5A	BF	5A	00	00	7B	85	BF	5A	15	00	89	09	00	00	¿Z¿Z...{¿Z¿Z...
0000C6A0	41	6C	00	4D	00	7A	00	45	00	75	00	0F	00	F1	00	00	A1.M.z.E.u...ñ..
0000C6B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy...yy
0000C6C0	4C	4D	5A	45	55	20	20	20	20	20	20	10	00	00	7B	85	LMZEU ...{
0000C6D0	BF	5A	BF	5A	00	00	7B	85	BF	5A	14	00	00	00	00	00	¿Z¿Z...{¿Z¿Z....
0000C6E0	41	43	00	48	00	67	00	69	00	75	00	0F	00	62	00	00	AC.H.g.i.u...b..
0000C6F0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF	yyyyyyyyyy...yy
0000C700	43	48	47	49	55	20	20	20	20	20	20	20	00	00	7D	85	CHGIU ..}
0000C710	BF	5A	BF	5A	00	00	7D	85	BF	5A	32	00	7D	03	00	00	¿Z¿Z...}¿Z¿Z...}
0000C720	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Номер сектора

99

Калькулятор

DEC: 79

HEX: 0x4f

Счеты

(0x0014-2)*1+61

lMzEu: (0x0014-2)*1+61=79

Здесь находим директорию «CHgiu»:

offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F		
00009E00	2E	20	20	20	20	20	20	20	20	20	20	10	00	00	7B	85		...{
00009E10	BF	5A	BF	5A	00	00	7B	85	BF	5A	14	00	00	00	00	00		¿Z¿Z...{¿Z¿Z.....
00009E20	2E	2E	20	20	20	20	20	20	20	20	20	10	00	00	7B	85	{
00009E30	BF	5A	BF	5A	00	00	7B	85	BF	5A	28	00	00	00	00	00		¿Z¿Z...{¿Z¿Z{.....
00009E40	41	43	00	48	00	67	00	69	00	75	00	0F	00	62	00	00		AC.H.g.i.u...b..
00009E50	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF		ýýýýýýýýýýýý..ýýýý
00009E60	43	48	47	49	55	20	20	20	20	20	20	10	00	00	7B	85		CHGIU ...{
00009E70	BF	5A	BF	5A	00	00	7B	85	BF	5A	7F	00	00	00	00	00		¿Z¿Z...{¿Z¿Z.....
00009E80	41	6C	00	4D	00	7A	00	45	00	75	00	0F	00	F1	00	00		A1.M.z.E.u...ñ..
00009E90	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF		ýýýýýýýýýýýý..ýýýý
00009EA0	4C	4D	5A	45	55	20	20	20	20	20	20	20	00	00	7B	85		LMZEU ...{
00009EB0	BF	5A	BF	5A	00	00	7B	85	BF	5A	45	00	61	07	00	00		¿Z¿Z...{¿Z¿Z.E.a...
00009EC0	48	51	44	4C	50	20	20	20	20	20	20	10	00	00	7B	85		HQDLP ...{
00009ED0	BF	5A	BF	5A	00	00	7B	85	BF	5A	13	00	00	00	00	00		¿Z¿Z...{¿Z¿Z.....
00009EE0	41	6C	00	55	00	76	00	57	00	65	00	0F	00	15	00	00		A1.U.v.W.e.....
00009EF0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF		ýýýýýýýýýýýý..ýýýý
00009F00	4C	55	56	57	45	20	20	20	20	20	20	20	00	00	7B	85		LUVWE ...{
00009F10	BF	5A	BF	5A	00	00	7B	85	BF	5A	36	00	B8	02	00	00		¿Z¿Z...{¿Z¿Z6....
00009F20	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

Номер сектора

79

Калькулятор

DEC:

186

HEX:

0xba

Счеты

(0x007f-2)*1+61

CHgiu: (0x007f-2)*1+61=186

В директории «CHgiu» по заданию должен лежать файл «lUvWe». Найдём его имя и удостоверимся, что это файл (*Атрибут* равен 20). Теперь вычислим *Размер файла в байтах*: для этого переведем шестнадцатиричную запись в десятичную. После воспользуемся формулой для нахождения *Количества кластеров для хранения файла*:

$$\left\lceil \frac{\text{Размер файла в байтах}}{\text{Размер сектора в байтах} \cdot \text{Размер кластера в секторах}} \right\rceil$$

Также запишем *Первый кластер файла*, чтобы не забыть о нём в будущем:

offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F		
00017400	2E	20	20	20	20	20	20	20	20	20	20	10	00	00	7B	85		...{
00017410	BF	5A	BF	5A	00	00	7B	85	BF	5A	7F	00	00	00	00	00		¿Z¿Z...{¿Z¿Z.....
00017420	2E	2E	20	20	20	20	20	20	20	20	20	10	00	00	7B	85	{
00017430	BF	5A	BF	5A	00	00	7B	85	BF	5A	14	00	00	00	00	00		¿Z¿Z...{¿Z¿Z.....
00017440	41	6C	00	4D	00	7A	00	45	00	75	00	0F	00	F1	00	00		A1.M.z.E.u...ñ..
00017450	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF		ýýýýýýýýýýýý..ýýýý
00017460	4C	4D	5A	45	55	20	20	20	20	20	20	20	00	00	7B	85		LMZEU ...{
00017470	BF	5A	BF	5A	00	00	7B	85	BF	5A	92	00	68	02	00	00		¿Z¿Z...{¿Z¿Z.h...
00017480	48	51	44	4C	50	20	20	20	20	20	20	20	00	00	7C	85		HQDLP ...{
00017490	BF	5A	BF	5A	00	00	7C	85	BF	5A	3E	00	8C	03	00	00		¿Z¿Z...{¿Z¿Z.....
000174A0	41	43	00	48	00	67	00	69	00	75	00	0F	00	62	00	00		AC.H.g.i.u...b..
000174B0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF		ýýýýýýýýýýýý..ýýýý
000174C0	43	48	47	49	55	20	20	20	20	20	20	20	00	00	7C	85		CHGIU ...{
000174D0	BF	5A	BF	5A	00	00	7C	85	BF	5A	38	00	85	05	00	00		¿Z¿Z...{¿Z¿Z.....
000174E0	41	6C	00	55	00	76	00	57	00	65	00	0F	00	15	00	00		A1.U.v.W.e.....
000174F0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	FF	FF	FF	FF		ýýýýýýýýýýýý..ýýýý
00017500	4C	55	56	57	45	20	20	20	20	20	20	20	00	00	7C	85		LUVWE ...{
00017510	BF	5A	BF	5A	00	00	7C	85	BF	5A	C4	00	4A	0A	00	00		¿Z¿Z...{¿Z¿Z.A.J...
00017520	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

Номер сектора

186

Калькулятор

DEC:

196

HEX:

0xc4

Счеты

0x00c4

lUvWe: Размер файла в байтах -> 4A 0A 00 00 -> 0x00000a4a = 2634 байт

Количество кластеров для хранения файла: 2634/(512*1) = 5.14453125 -> 6 кластеров

Первый кластер файла: 0x00c4 = 196

Если до этого момента всё было правильно, то почти вся оставшаяся работа будет проходить только с FAT-таблицей, поэтому перемещаемся к первой таблице: для этого надо пропустить все зарезервированные секторы, то есть в поле «Номер сектора» пишем *Количество зарезервированных секторов*. Сразу же под словом offset находится значение *Смещения FAT-таблицы* в шестнадцатиричной записи. Вынесем это значение в поле «Для заметок»:

offset	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000A00	F8	FF	FF	00	B0	00	4A	C0	03	D9	00	00	17	A0	08	00	øÿÿ.°.JÀ.Û... ..
00000A10	E0	06	00	E0	02	00	F0	FF	63	F0	FF	00	F0	FF	FF	9F	à...à...ðÿcðÿ.ðÿÿð
00000A20	02	00	80	07	19	F0	FF	00	00	04	00	F0	FF	00	30	02	..ð...ðÿ....ðÿ.0.
00000A30	27	E0	04	00	40	03	00	00	00	00	C0	02	FF	AF	05	00	'à...@.....À.ÿ'..
00000A40	F0	08	55	F0	FF	35	00	00	6D	50	06	A0	00	00	50	B0	ð.Uðÿ5...mP. ...P°
00000A50	09	54	00	00	39	E0	05	4D	00	00	62	F0	FF	97	00	00	.T...9à.M...bðÿð..
00000A60	61	F0	0B	71	00	00	87	40	07	76	00	00	00	00	00	8E	að.q...ð@.v.....ð
00000A70	00	00	FF	0F	08	FF	0F	00	70	20	07	FF	0F	00	FF	6F	..ÿ...ÿ...p .ÿ...ÿo
00000A80	05	A7	60	09	00	10	08	8D	00	00	00	50	08	FF	FF	FF	.§`....ð...P.ÿÿÿ
00000A90	00	90	09	68	20	0A	00	B0	0C	69	00	00	82	E0	0A	86	.ð.h ..°.i...ðà.ð
00000AA0	00	00	B0	30	09	FF	0F	00	A4	50	0D	B9	00	00	D8	00	..°0.ÿ...mP.¹...0.
00000AB0	00	AB	00	00	FD	40	08	B2	00	00	C0	00	00	00	F0	FF	.«...ÿ@.²...À...ðÿ
00000AC0	91	70	0B	FF	0F	00	E0	D0	0B	FF	9F	08	00	D0	0C	CF	ðp.ÿ...àð.ÿð.ð.İ
00000AD0	70	0E	00	90	11	AC	C0	0B	00	A0	09	EE	50	09	00	80	p...ð.¬À... .iP...ð
00000AE0	0A	FF	FF	FF	00	30	0B	06	F1	0A	DB	00	00	FF	CF	0D	.ÿÿÿ.0...ñ.Û...ÿİ.
00000AF0	FF	0F	00	AA	00	00	FF	0F	00	00	F0	FF	B8	00	00	CC	ÿ...²...ÿ...ðÿ...İ
00000B00	F0	FF	E9	00	00	B4	40	0D	02	01	00	FF	9F	0C	E5	00	ðÿé...´@....ÿð.â.
00000B10	00	C8	D0	0D	FF	5F	10	00	F0	FF	D0	E0	0D	00	10	0C	.Ëð.ÿ...ðÿðà....
00000B20	D3	60	0E	00	00	00	C7	E0	0F	00	70	0D	FF	FF	FF	00	Ó`....Çà...p.ÿÿÿ.
00000B30	D0	0E	FF	AF	0E	00	F0	FF	FF	FF	0F	00	F0	FF	FF	FF	ð.ÿ`...ðÿÿÿ...ðÿÿÿ
00000B40	FF	00	10	0F	FA	F0	FF	00	60	0F	E2	30	0F	FF	0F	00	ÿ...úðÿ.`.à0.ÿ..
00000B50	FF	0F	00	FF	0F	00	00	F0	FF	EB	F0	0E	00	70	10	FF	ÿ..ÿ...ðÿéð...p.ÿ
00000B60	5F	0F	00	F0	FF	FF	FF	FF	00	70	0F	00	F0	FF	00	F0	_...ðÿÿÿÿ.p...ðÿ.ð
00000B70	FF	03	91	0F	00	F0	FF	FF	0F	00	00	00	10	FF	8F	10	ÿ.ð...ðÿÿ....ÿð.
00000B80	1B	01	00	FF	FF	FF	00	F0	FF	FF	FF	FF	FF	0F	00	00	...ÿÿÿ.ðÿÿÿÿÿ...
00000B90	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000BA0	00	00	00	00	00	F0	FF	00	F0	FF	00	00	00	00	00	00ðÿ.ðÿ.....
00000BB0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000BC0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	60`
00000BD0	14	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000BE0	00	00	00	00	00	00	00	00	00	FF	0F	00	00	00	00	00ÿ.....
00000BF0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Номер сектора	5
---------------	---

Калькулятор

DEC:	196	HEX:	0xc4
------	-----	------	------

Счеты	0x00c4
-------	--------

Смещение FAT-таблицы - 0xA00 = 2560

Так как мы работаем с FAT12, то в FAT-таблице каждый кластер занимает по 1,5 байта. Воспользуемся формулой для нахождения кластера в FAT-таблице:

Смещение FAT-таблицы + *Номер кластера* · *Размер кластера в FAT-таблице*.

Здесь вместо Номер кластера подставляем значение, посчитанное на позапрошлом этапе, то есть 196. У нас получилось целое число. Это значит, что после того, как мы найдём байт со значениями строки и столбца, равным 0xb20 и 0x06, соответственно, мы возьмем ещё 2 байта справа от найденного. А также после «разворота» и деления пополам, мы возьмём правую часть получившейся записи. Выбранная часть указывает на следующий кластер файла:

offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

00000A00 F8 FF FF 00 B0 00 4A C0 03 D9 00 00 17 A0 08 00| øÿÿ.°.ÿÀ.Û... ..

00000A10 E0 06 00 E0 02 00 F0 FF 63 F0 FF 00 F0 FF FF 9F| à.à.äÿсäÿ.äÿÿ

00000A20 02 00 80 07 19 F0 FF 00 00 04 00 F0 FF 00 30 02| ..â.äÿ....äÿ.ø.

00000A30 27 E0 04 00 40 03 00 00 00 C0 02 FF AF 05 00| 'à.ä.ä....Ä.ÿ~..

00000A40 F0 08 55 F0 FF 35 00 00 6D 50 06 A0 00 00 50 B0| ä.Uäÿ5..мР. ..Р°

00000A50 09 06 00 00 39 E0 05 4D 00 00 62 F0 FF 97 00 00| .Т..9ä.М..bäÿ

00000A60 61 F0 0B 71 00 00 87 40 07 76 00 00 00 00 8E| аä.q.ä@.v.....ä

00000A70 00 00 FF 0F 08 FF 0F 00 70 20 07 FF 0F 00 FF 6F| ..ÿ..ÿ..р.ÿ..ÿø

00000A80 05 A7 60 09 00 10 08 8D 00 00 00 50 08 FF FF FF| .ç`....â...Р.ÿÿÿ

00000A90 00 90 09 68 20 0A 00 B0 0C 69 00 00 82 E0 0A 86| .â.h ..°.i...ââ.â

00000AA0 00 00 B0 30 09 FF 0F 00 A4 50 0D B9 00 00 D8 00| ..°ø.ÿ..мР.¹..ø.

00000AB0 00 A8 00 00 FD 40 03 B2 00 00 C0 00 00 00 F0 FF| «...ÿ@.²...Ä...äÿ

00000AC0 91 70 0B FF 0F 00 E0 D0 0B FF 9F 08 00 D0 0C CF| âр.ÿ..äð.ÿâ..ð.İ

00000AD0 70 0E 00 90 11 AC C0 08 00 A0 09 EE 50 09 00 80| р..â..Ä...iР..â

00000AE0 0A FF FF FF 00 30 03 06 F1 0A DB 00 00 FF CF 0D| .ÿÿÿ.ø..ñ.Û..ÿİ.

00000AF0 FF 0F 00 AA 00 00 FF 0F 00 00 F0 FF B8 00 00 CC| ÿ..ä..ÿ...äÿ...İ

00000B00 F0 FF E9 00 00 B4 40 0D 02 01 00 FF 9F 0C E5 00| äÿé...`@...ÿâ.ä.

00000B10 00 C8 D0 0D FF 5F 10 00 F0 FF D0 E0 0D 00 10 0C| .èð.ÿ...äÿðâ....

00000B20 03 60 0E 00 00 00 C7 E0 0F 00 70 0D FF FF FF 00| ó`....Çä..р.ÿÿÿ.

00000B30 D0 0E FF AF 0E 00 F0 FF FF FF 0F 00 F0 FF FF FF| ð.ÿ`..äÿÿÿ..äÿÿÿ

00000B40 FF 00 10 0F FA F0 FF 00 60 0F E2 30 0F FF 0F 00| ÿ...úäÿ..`äø.ÿ..

00000B50 FF 0F 00 FF 0F 00 00 F0 FF EB F0 0E 00 70 10 FF| ÿ..ÿ...äÿéð..р.ÿ

00000B60 5F 0F 00 F0 FF FF FF FF 00 70 0F 00 F0 FF 00 F0| _..äÿÿÿÿ.р..äÿ.ä

00000B70 FF 03 91 0F 00 F0 FF FF 0F 00 00 00 10 FF 8F 10| ÿ.â..äÿÿ....ÿâ.

00000B80 1B 01 00 FF FF FF FF 00 F0 FF FF FF FF FF 0F 00 00| ...ÿÿÿ.äÿÿÿÿÿ...

00000B90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|

00000BA0 00 00 00 00 00 00 F0 FF 00 F0 FF 00 00 00 00 00|äÿ.äÿ.....

00000BB0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|

00000BC0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 60|`

00000BD0 14 00 00 00 00 00 00 00 00 00 00 00 00 00 00|

00000BE0 00 00 00 00 00 00 00 00 00 FF 0F 00 00 00 00 00|ÿ.....

00000BF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|

Калькулятор

DEC: 199

HEX: 0xc7

Счеты 0x0c7

1 кластер: $2560 + 196 * 1.5 = 2854 = 0xb26 \Rightarrow b2$ и 6
целое $\Rightarrow C7 E0 0F \rightarrow 0fe|0c7 \Rightarrow 0x0c7=199$

Снова подставляем значения в формулу. Теперь Номер кластера равен 199. Значение получилось дробным. Значит, округляем значение вниз до целого значения и послетого, как найдём байт в таблице, возьмём ещё «соседей», а также после «разворота» возьмём левую часть записи. Получаем число, указывающее на следующий кластер файла:

offset 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F

00000A00 F8 FF FF 00 B0 00 4A C0 03 D9 00 00 17 A0 08 00| øÿÿ.°.ÿÀ.Û... ..

00000A10 E0 06 00 E0 02 00 F0 FF 63 F0 FF 00 F0 FF FF 9F| à.à.äÿсäÿ.äÿÿ

00000A20 02 00 80 07 19 F0 FF 00 00 04 00 F0 FF 00 30 02| ..â.äÿ....äÿ.ø.

00000A30 27 E0 04 00 40 03 00 00 00 C0 02 FF AF 05 00| 'à.ä.ä....Ä.ÿ~..

00000A40 F0 08 55 F0 FF 35 00 00 6D 50 06 A0 00 00 50 B0| ä.Uäÿ5..мР. ..Р°

00000A50 09 54 00 00 39 E0 05 4D 00 00 62 F0 FF 97 00 00| .Т..9ä.М..bäÿ

00000A60 61 F0 0B 71 00 00 87 40 07 76 00 00 00 00 8E| аä.q.ä@.v.....ä

00000A70 00 00 FF 0F 08 FF 0F 00 70 20 07 FF 0F 00 FF 6F| ..ÿ..ÿ..р.ÿ..ÿø

00000A80 05 A7 60 09 00 10 08 8D 00 00 00 50 08 FF FF FF| .ç`....â...Р.ÿÿÿ

00000A90 00 90 09 68 20 0A 00 B0 0C 69 00 00 82 E0 0A 86| .â.h ..°.i...ââ.â

00000AA0 00 00 B0 30 09 FF 0F 00 A4 50 0D B9 00 00 D8 00| ..°ø.ÿ..мР.¹..ø.

00000AB0 00 AB 00 00 FD 40 03 B2 00 00 C0 00 00 00 F0 FF| «...ÿ@.²...Ä...äÿ

00000AC0 91 70 0B FF 0F 00 E0 D0 0B FF 9F 08 00 D0 0C CF| âр.ÿ..äð.ÿâ..ð.İ

00000AD0 70 0E 00 90 11 AC C0 08 00 A0 09 EE 50 09 00 80| р..â..Ä...iР..â

00000AE0 0A FF FF FF 00 30 03 06 F1 0A DB 00 00 FF CF 0D| .ÿÿÿ.ø..ñ.Û..ÿİ.

00000AF0 FF 0F 00 AA 00 00 FF 0F 00 00 F0 FF B8 00 00 CC| ÿ..ä..ÿ...äÿ...İ

00000B00 F0 FF E9 00 00 B4 40 0D 02 01 00 FF 9F 0C E5 00| äÿé...`@...ÿâ.ä.

00000B10 00 C8 D0 0D FF 5F 10 00 F0 FF D0 E0 0D 00 10 0C| .èð.ÿ...äÿðâ....

00000B20 03 60 0E 00 00 00 C7 E0 0F 00 70 0D FF FF FF 00| ó`....Çä..р.ÿÿÿ.

00000B30 D0 0E FF AF 0E 00 F0 FF FF FF 0F 00 F0 FF FF FF| ð.ÿ`..äÿÿÿ..äÿÿÿ

00000B40 FF 00 10 0F FA F0 FF 00 60 0F E2 30 0F FF 0F 00| ÿ...úäÿ..`äø.ÿ..

00000B50 FF 0F 00 FF 0F 00 00 F0 FF EB F0 0E 00 70 10 FF| ÿ..ÿ...äÿéð..р.ÿ

00000B60 5F 0F 00 F0 FF FF FF FF FF 00 70 0F 00 F0 FF 00 F0| _..äÿÿÿÿ.р..äÿ.ä

00000B70 FF 03 91 0F 00 F0 FF FF 0F 00 00 00 10 FF 8F 10| ÿ.â..äÿÿ....ÿâ.

00000B80 1B 01 00 FF FF FF FF 00 F0 FF FF FF FF FF 0F 00 00| ...ÿÿÿ.äÿÿÿÿÿ...

00000B90 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|

00000BA0 00 00 00 00 00 00 F0 FF 00 F0 FF 00 00 00 00 00|äÿ.äÿ.....

00000BB0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|

00000BC0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 60|`

00000BD0 14 00 00 00 00 00 00 00 00 00 00 00 00 00 00|

00000BE0 00 00 00 00 00 00 00 00 00 FF 0F 00 00 00 00 00|ÿ.....

00000BF0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00|

Калькулятор

DEC: 215

HEX: 0xd7

Счеты 0x0d7

2 кластер: $2560 + 199 * 1.5 = 2858.5 \rightarrow 2858 = 0xb2a \Rightarrow b2$ и а
дробное $\Rightarrow 00 70 0D \rightarrow 0d7|000 \Rightarrow 0x0d7=215$

Повторяем эти действия до тех пор, пока не получим значение 0xFFFF. Когда это произойдёт, то у нас будут данные о всех кластерах, что хранят информацию о файле:

[illegible][illegible][illegible][illegible]

Теперь подставляем все имеющиеся данные о кластерах в формула для перехода к нужному кластеру:

$$(\text{Номер кластера} - 2) \cdot \text{Размер кластера в секторах} + \text{Начало области данных}$$

Имеющиеся данные:

1 кластер: $(196-2)*1+61=255$

2 кластер: $(199-2)*1+61=258$

3 кластер: $(215-2)*1+61=274$

4 кластер: $(241-2)*1+61=300$









5 кластер: $(247-2)*1+61=306$

6 кластер: $(249-2)*1+61=308$

Переходим по всем этим кластерам и выгружаем их. Так как у нас *Размер кластера в секторах* равен 1, то на каждый кластер скачиваем только 1 сектор:

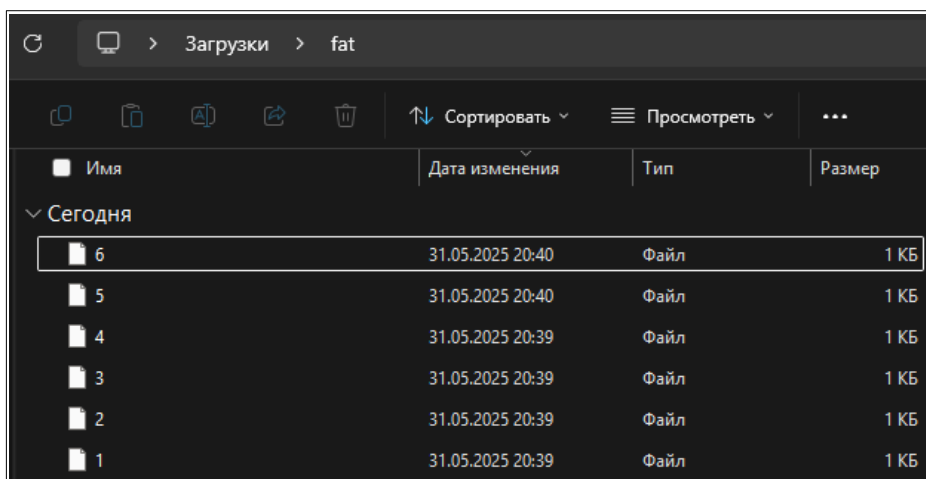
[illegible][illegible][illegible][illegible][illegible][illegible]

Найдём выгруженные секторы в «Загрузках» с помощью «Проводника»:

Сегодня			
	1db8429f-349d-4b10-8963-43616bd9e...	31.05.2025 20:40	1 КБ
	ac17919f-4931-4db6-86fd-c8702ec8a9...	31.05.2025 20:40	1 КБ
	0416c41b-b4de-4d76-b81b-02660843f...	31.05.2025 20:39	1 КБ
	bdf3d9cf-8a3d-496c-9d04-0748b14db...	31.05.2025 20:39	1 КБ
	8197c385-52ac-4d1a-a30f-0803c44bcc...	31.05.2025 20:39	1 КБ
	3b1582f7-c059-40a8-8a9f-2d0030134a89	31.05.2025 20:39	1 КБ
	Telegram Desktop	31.05.2025 19:06	Папка с файлами
	fat	31.05.2025 17:49	Папка с файлами

Создадим новую папку и переместим в неё скачанные файлы, чтобы нам ничего другого не мешало:

Переименуем файлы в порядке скачивания, используя для удобства натуральные числа в качестве имён:



Имя	Дата изменения	Тип	Размер
6	31.05.2025 20:40	Файл	1 КБ
5	31.05.2025 20:40	Файл	1 КБ
4	31.05.2025 20:39	Файл	1 КБ
3	31.05.2025 20:39	Файл	1 КБ
2	31.05.2025 20:39	Файл	1 КБ
1	31.05.2025 20:39	Файл	1 КБ

Перейдём в «Терминал» и выполним следующие команды:

- Для объединения файлов в один:

cat список имён файлов через пробелы (от 1 до последнего) > любое имя файла

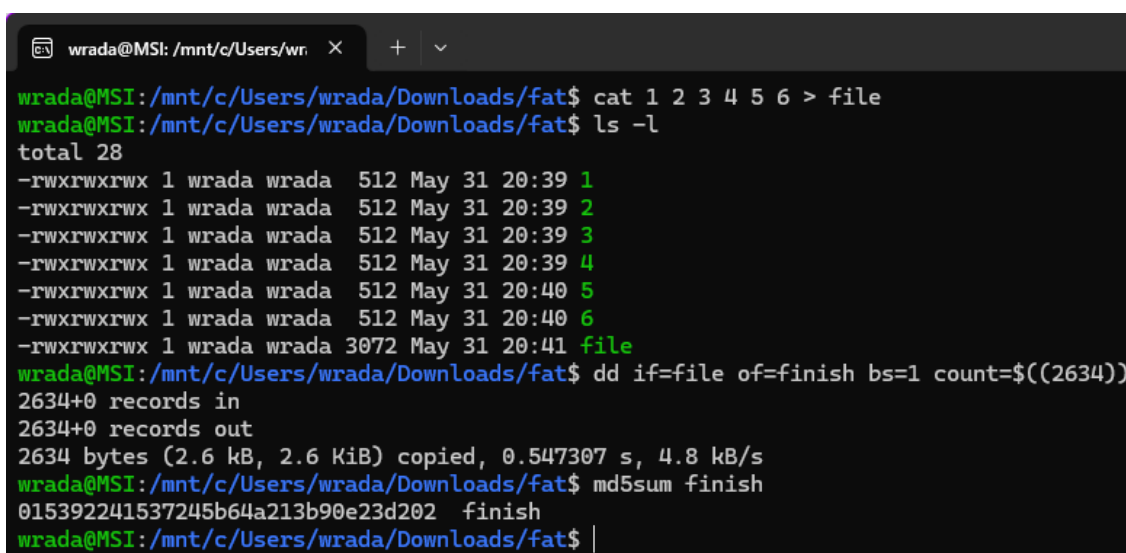
- Для изменения размера файла, где Размер файла в байтах берём с этапа, где мы только нашли файл на сайте:

dd if=имя объединенного файла of=любое имя файла bs=1 count=\$((размер файла в байтах))

- Для подсчёта контрольной суммы:

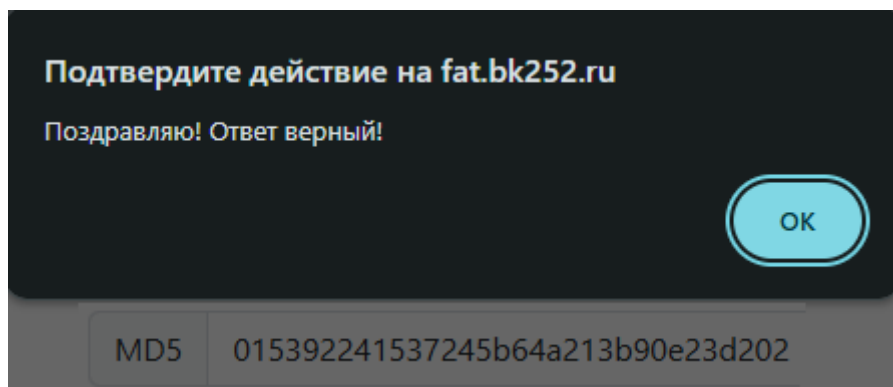
md5sum имя файла из прошлой команды

Пример использования команд для нашей задачи:



```
wrada@MSI: /mnt/c/Users/wrada/Downloads/fat$ cat 1 2 3 4 5 6 > file
wrada@MSI: /mnt/c/Users/wrada/Downloads/fat$ ls -l
total 28
-rwxrwxrwx 1 wrada wrada 512 May 31 20:39 1
-rwxrwxrwx 1 wrada wrada 512 May 31 20:39 2
-rwxrwxrwx 1 wrada wrada 512 May 31 20:39 3
-rwxrwxrwx 1 wrada wrada 512 May 31 20:39 4
-rwxrwxrwx 1 wrada wrada 512 May 31 20:40 5
-rwxrwxrwx 1 wrada wrada 512 May 31 20:40 6
-rwxrwxrwx 1 wrada wrada 3072 May 31 20:41 file
wrada@MSI: /mnt/c/Users/wrada/Downloads/fat$ dd if=file of=finish bs=1 count=$((2634))
2634+0 records in
2634+0 records out
2634 bytes (2.6 kB, 2.6 KiB) copied, 0.547307 s, 4.8 kB/s
wrada@MSI: /mnt/c/Users/wrada/Downloads/fat$ md5sum finish
015392241537245b64a213b90e23d202 finish
wrada@MSI: /mnt/c/Users/wrada/Downloads/fat$ |
```


Копируем вывод последней команды без имени файла и вставляем на сайте в поле md5. Отправляем на проверку и получаем уведомление в верхней части экрана:



ФАЙЛОВАЯ СИСТЕМА NTFS