

**МИНОБРНАУКИ РОССИИ**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технологический университет «СТАНКИН»**

**(ФГБОУ ВО «МГТУ «СТАНКИН»)**

Институт цифровых интеллектуальных систем  
Кафедра компьютерных систем управления

Образовательная программа 15.03.04

«Автоматизация технологических процессов и производств»

Дисциплина «Основы системного программного обеспечения»

**Отчет по лабораторной работе №2**

Выполнил:

студент гр. АДБ-21-08 \_\_\_\_\_\_\_\_\_\_\_\_ Герасимов Д.С.

(дата) (подпись)

Принял:

к.т.н., доцент \_\_\_\_\_\_\_\_\_\_\_\_ Ковалев И.А.

(дата) (подпись)

**Москва 2024**

**Цель работы:** овладеть навыками работы с консольной утилитой DiskPart для управления разделами жестких дисков, навыками редактирования секторов жестких дисков и логических разделов, а также изучить основы работы с командной строкой Windows.

**Практическая часть**

Ссылка на репозиторий к лабораторной работе: https://github.com/sselfassured/GerasimovDS\_LR2

Diskpart — консольная утилита для управления разделами жестких дисков. Впервые появилась в линейке операционных систем Windows NT, начиная с Windows 2000, заменив собой fdisk, который использовался в основанных на MS-DOS операционных системах.

**ЗАДАНИЕ 1.** Попробуйте все перечисленные выше команды, посмотрите, чем они отличаются. Ниже пример выполнения **list disk** в программе diskpart.

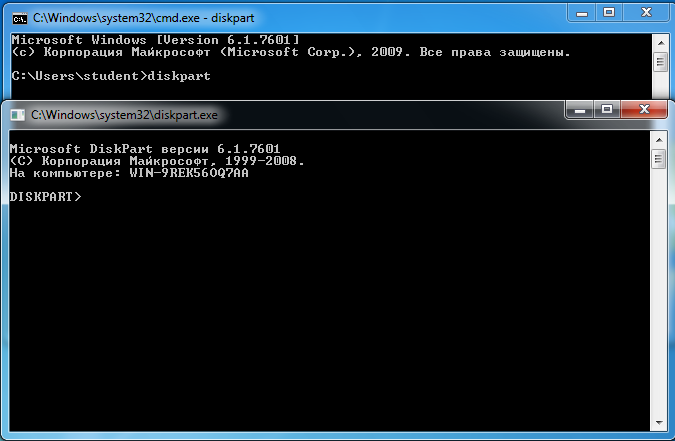


Рис. 1. Запущенная консольная утилита DiskPart.

Для перечисления доступных дисков, разделов и томов предназначены команды:

**list disk** – перечисляет все физические жесткие диски компьютера;

**list volume** – перечисляет все дисковые тома (включая разделы жесткого диска и логические диски);

**list partition** – перечисляет разделы на диске, который находится в фокусе;

**select** – устанавливает фокус на объект.

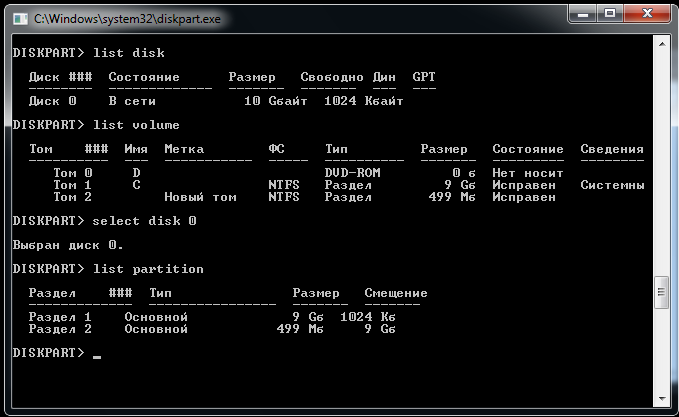


Рис. 2. Выполненные команды для перечисления доступных дисков, томов и разделов.

**ЗАДАНИЕ 2.** Установить имя тому с размером 500 MB в соответствии с первой буквой Вашей фамилии.

Для выполнения этого задания нужно воспользоваться командой **assign letter k**, где вместо k поставить первую букву фамилии (G).

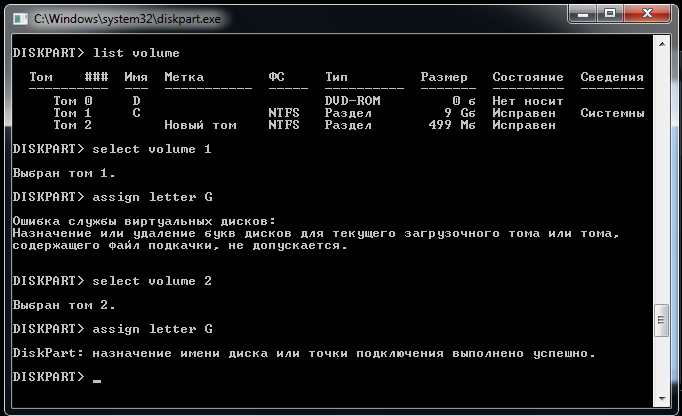


Рис. 3. Присвоение тому 2 размером 499 Мб имени.

**ЗАДАНИЕ 3.** Отформатируйте раздел 500MB в fat32, а после снова в ntfs. После каждого раза сохраняйте скриншоты. Убедитесь, что у вас выбран именно том с размером 500 MB (стоит звездочка напротив имени тома) и потом используйте команды format fs=fat32 и fs=ntfs. Для такого маленького диска можно использовать полноценное форматирование. Либо добавить в конец команды quick – для быстрого форматирования.

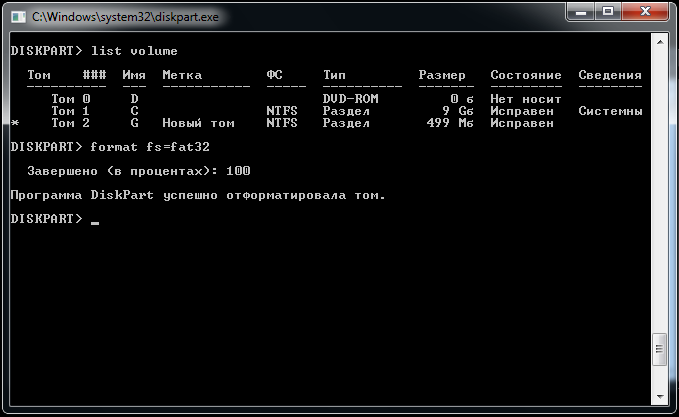


Рис. 4. Форматирование тома с именем G в файловую систему FAT32.

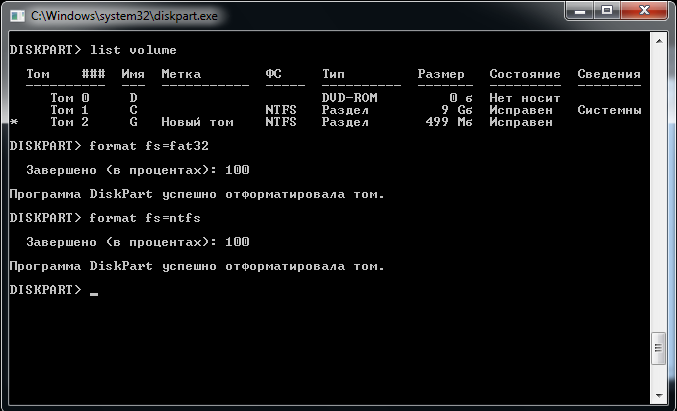


Рис. 5. Форматирование тома с именем G в файловую систему NTFS.

**ЗАДАНИЕ 4.** Разбить диск 500 MB на диски по 300 и 200 МB. После чего снова объединить их обратно.

Для выполнения этого задания нужно воспользоваться командой **shrink [desired=<desiredsize>] [minimum=<minimumsize>]**, где вместо <**desiredsize>** необходимо подставить 300, а вместо <**minimumsize>** 200.

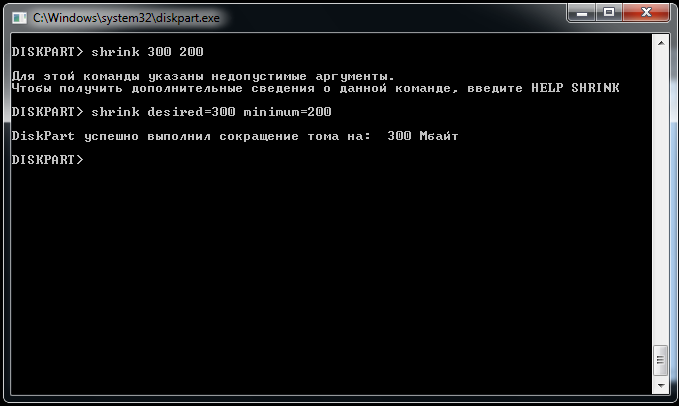


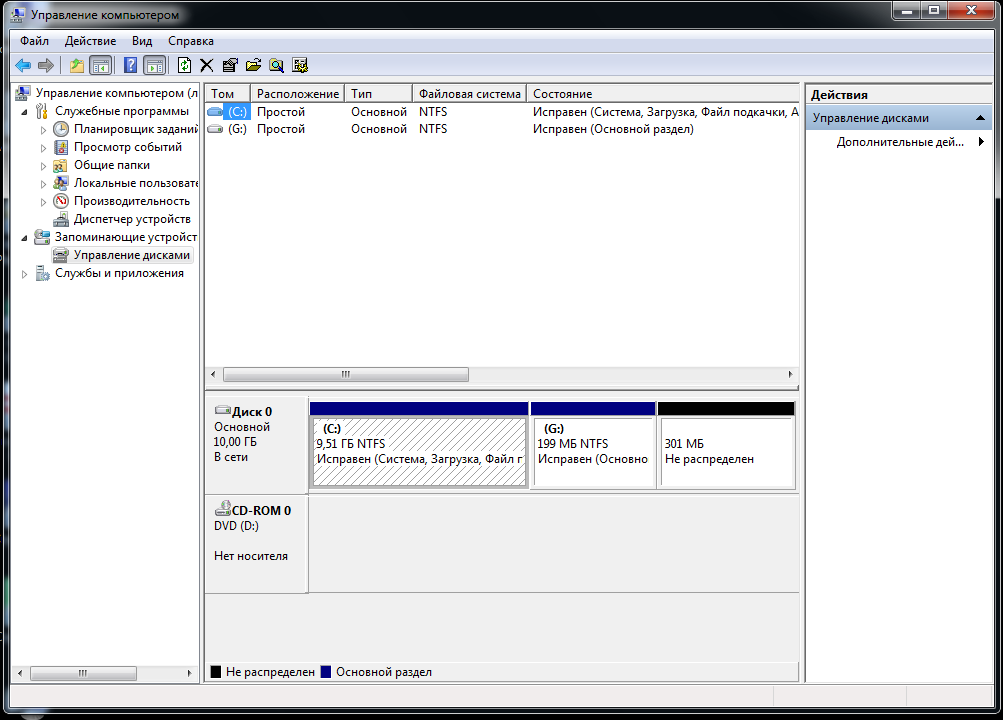
Рис. 6. Сжатие тома с именем G на 300 Мб.

Рис. 7. Отображение томов диска 0 в «Управлении дисками».

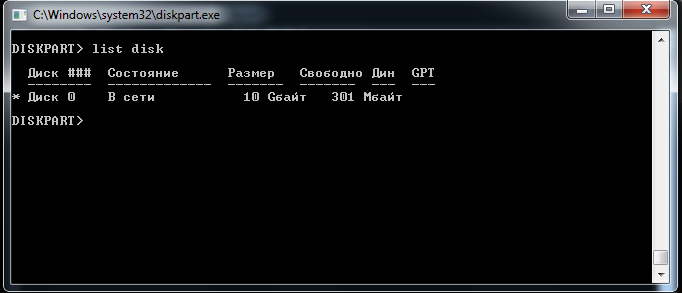


Рис. 8. Команда list disk указывает на неразмеченное пространство диска 0.

Воспользуемся командой **create partition primary** для создания нового тома на диске 0 на неразмеченном пространстве.

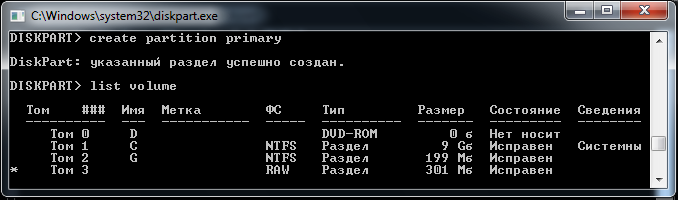


Рис. 9. Создание нового тома.

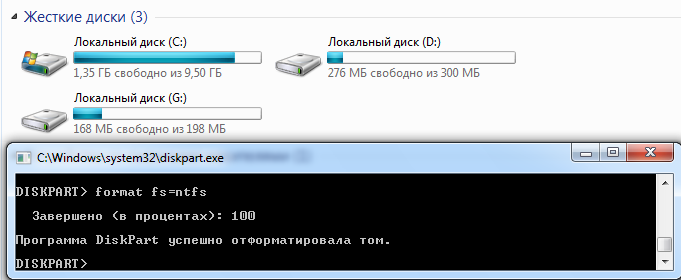


Рис. 10. Форматирование нового тома в ФС NTFS.

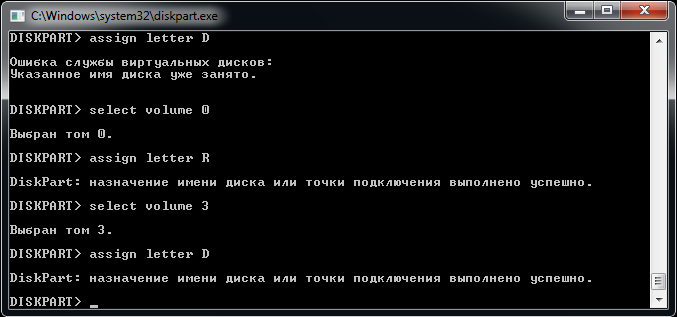


Рис. 11. Назначение тому имени.

**ЗАДАНИЕ 5.** Вернуть все как было.

Удаляем тома 301 MB и 199 MB используя команду **delete.**

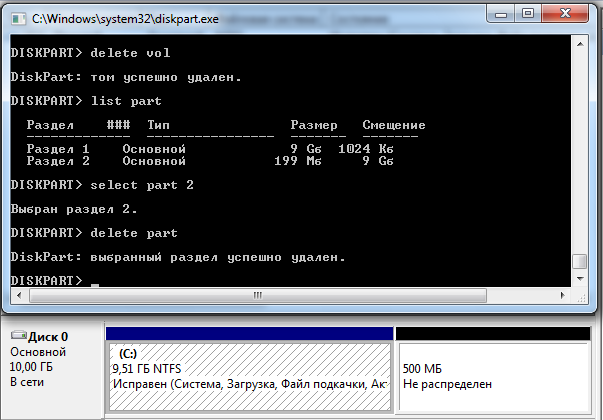


Рис. 12. Удаление разделов 301 Мб и 199 Мб.

Воспользуемся командой **create partition primary** для создания нового тома на диске 0 на неразмеченном пространстве.

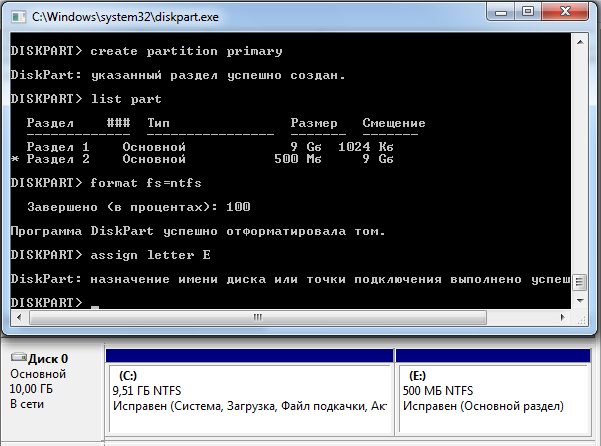


Рис. 13. Создание нового раздела на диске 0, форматирование его в NTFS и присвоение имени.

**ЗАДАНИЕ 6.** Сделаем раздел с загрузчиком неактивным.

Для выполнения этого задания нужно воспользоваться командой **detail part**, которая отображает свойства выбранного раздела. Далее командой **inactive** нужно сделать раздел неактивным.

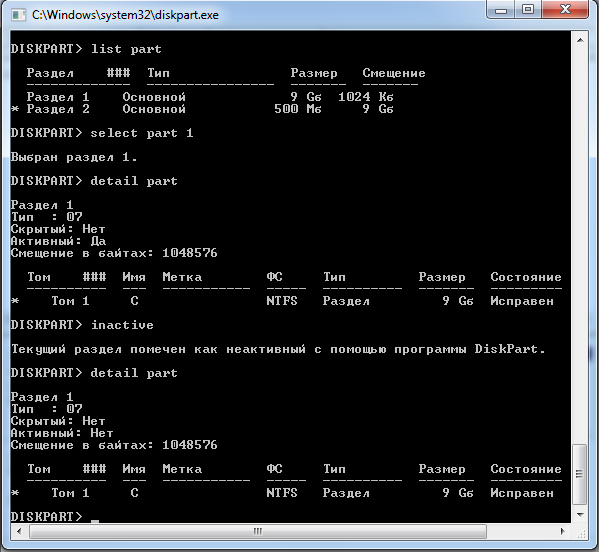


Рис. 14. Командой **inactive** раздел с загрузчиком стал неактивным.

После перезагрузки ОС не загружается, так как в таблицу разделов нет активного тома, где располагается загрузчик.

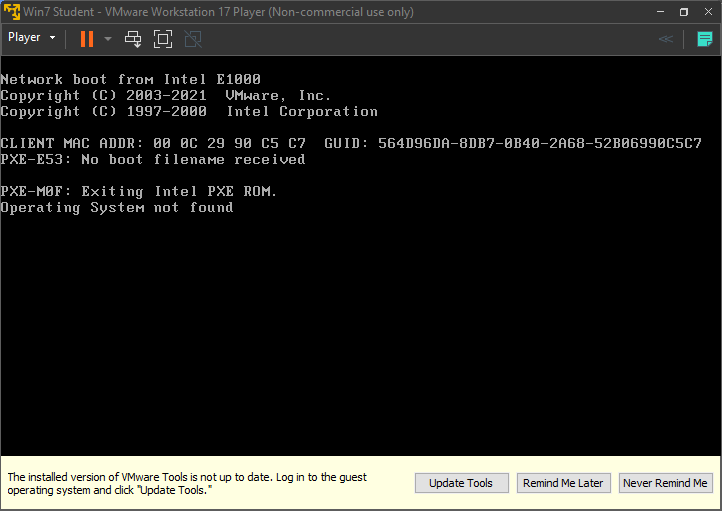


Рис. 15. Операционная система не найдена.

**ЗАДАНИЕ 6.** Сделаем раздел с загрузчиком активным.

С помощью файла с образом Windows переходим в установку Windows, где доступна командная строка. Запускаем DiskPart.

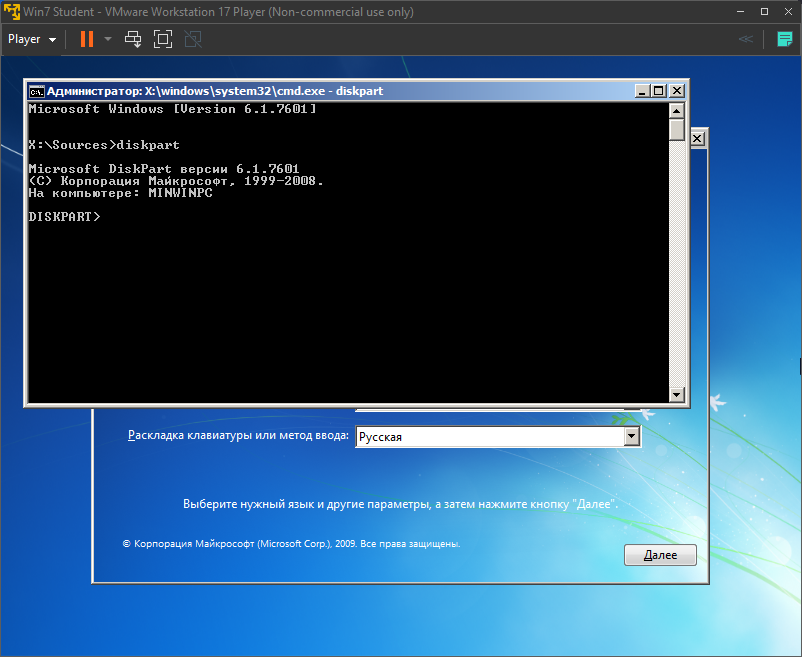


Рис. 16. Запущенная утилита DiskPart в командной строке из инсталлятора Windows.

С помощью команды **active** делаем том с загрузчиком снова активным.

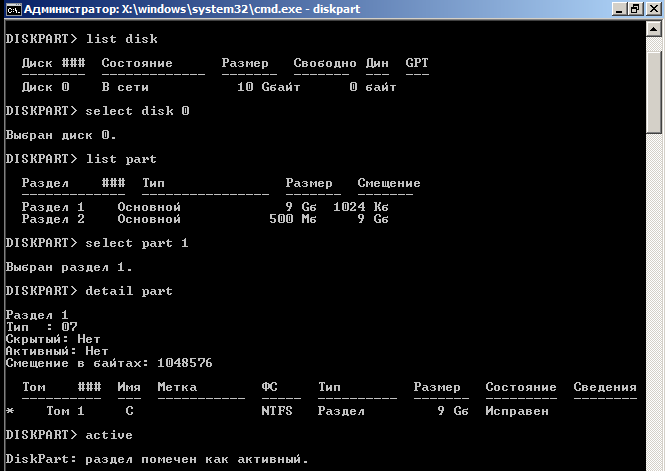


Рис. 17. Активизация тома с загрузчиком.

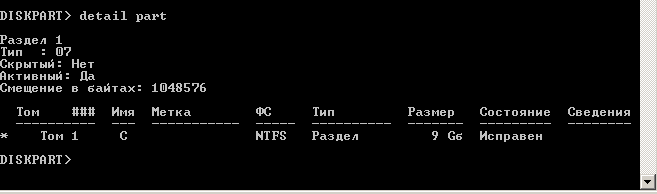


Рис. 18. Свойства активированного тома.

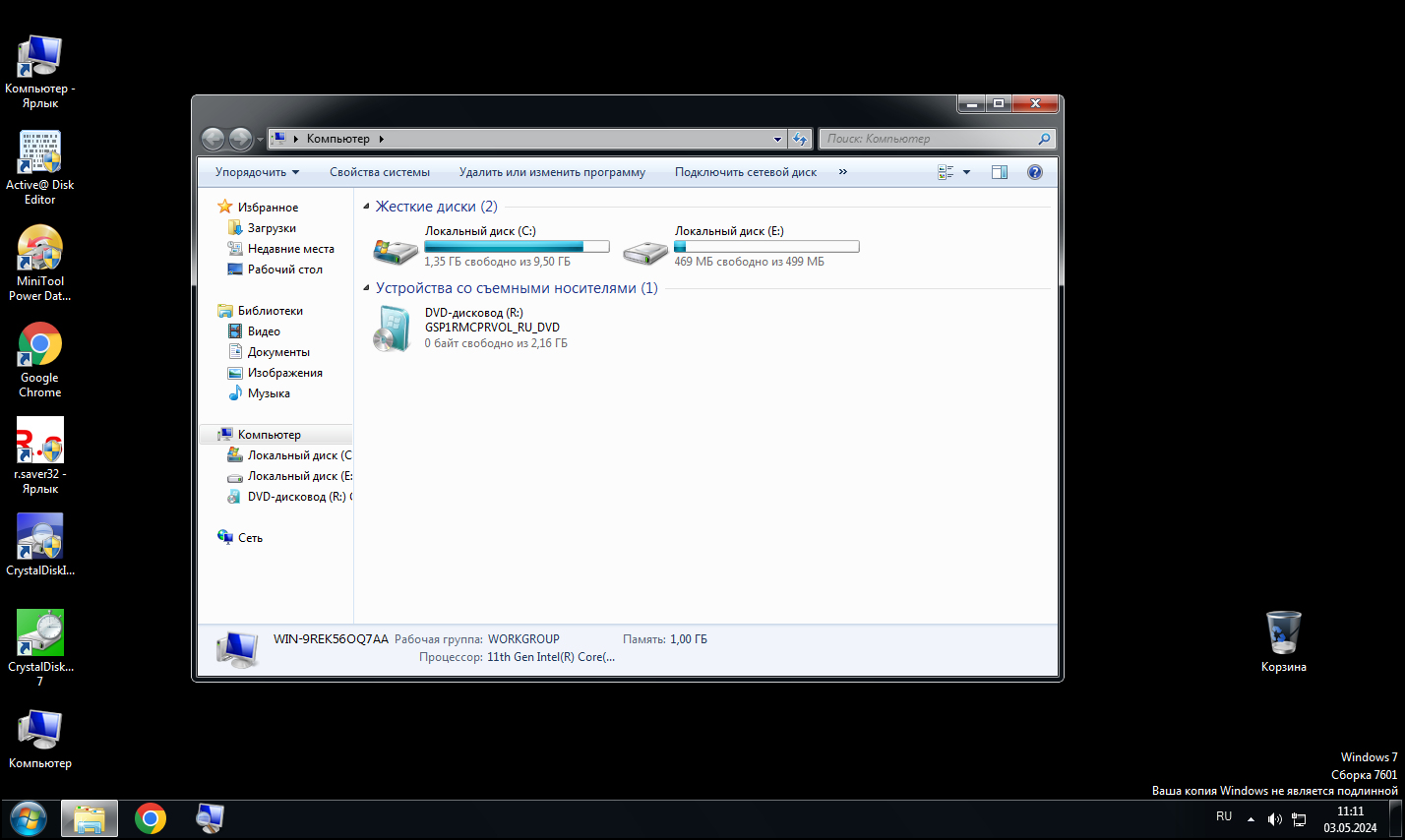


Рис. 19. Успешный запуск Windows.

**ЗАДАНИЕ 7.** Сломаем загрузчик.

Для выполнения данного задания нужна программа Active@ Disk Editor. Active@ Disk Editor — это программа, которая позволяет редактировать сектора жестких дисков и логических разделов. При её открытии в шестнадцатеричном формате представлен первый сектор жесткого диска. Пространство, выделенное вначале одним цветом, называется boot код или загрузчик. Этот код позволяет найти активный раздел и передать управления первому сектору в нем. Далее идет таблица разделов, последние буквы (55AA) означают конец главной загрузочной записи.

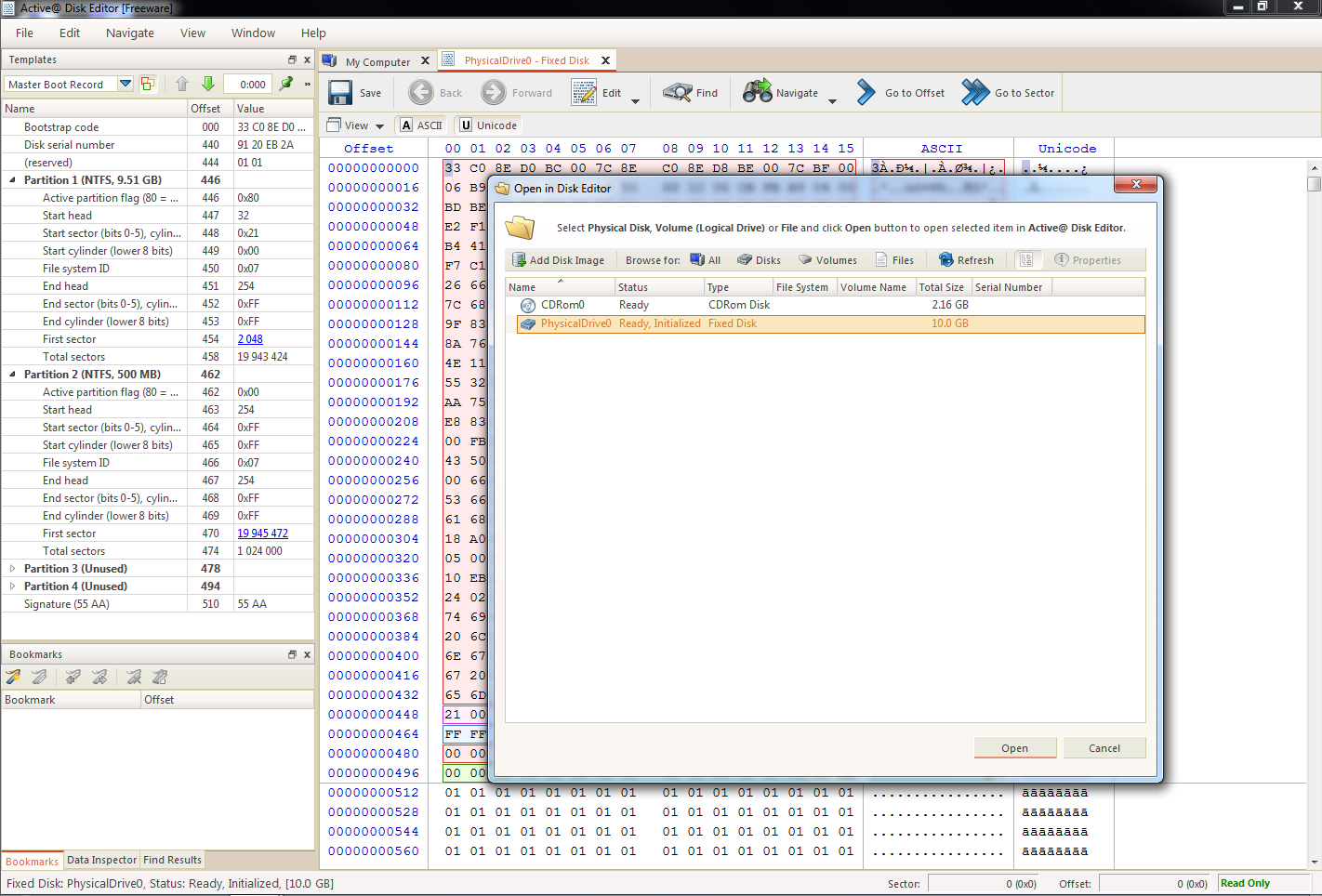


Рис. 20. Запущенная программа Active@ Disk Editor.

Произвольно меняем boot код:

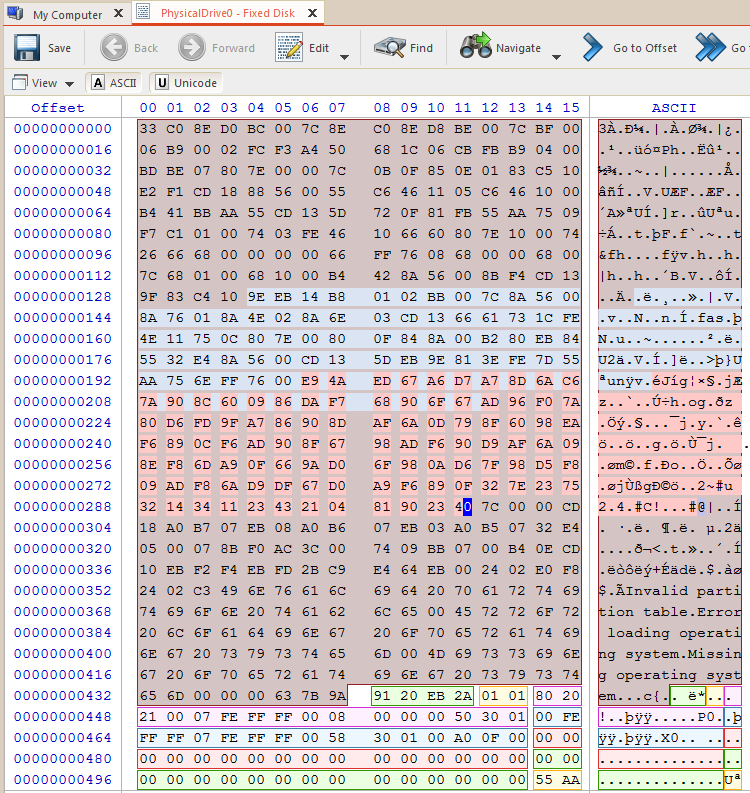


Рис. 21. Редактирование загрузочного кода.

После этих манипуляций загрузчик «сломан». При попытке перезагрузки Windows уже не загружается.

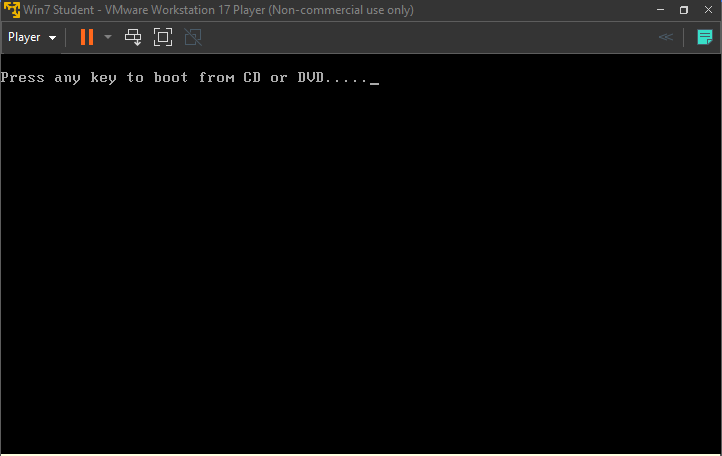


Рис. 22. Попытка загрузки в ОС при «сломанном» загрузчике.

**ЗАДАНИЕ 8.** Восстановим загрузчик.

Проделываем такие же действия, как и в задании 6. Запускаем DiskPart и смотрим, на каком томе у нас загрузчик.

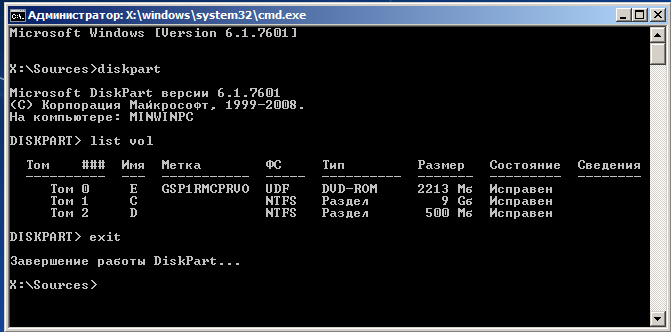


Рис. 23. Запуск и выход из утилиты DiskPart.

Для восстановления загрузчика воспользуемся утилитой bootsect.exe.

Утилита BOOTSECT.EXE позволяет изменить программный код загрузчика Windows для переключения между двумя вариантами диспетчера загрузки - BOOTMGR или NTLDR.

Команда BOOTSECT позволяет записывать заданный программный код загрузочных секторов, обеспечивающих загрузку либо ntldr, либо bootmgr.

Формат командной строки:

**bootsect {/help|/nt60|/nt52} {SYS|ALL|< DriveLetter >:} [/force] [/mbr]**

Параметры командной строки Bootsect:

**/help** - отображение справочной информации;

/nt52 - запись программного кода загрузочного сектора, обеспечивающего использование загрузчика ntldr для операционных систем, предшествующих Windows Vista.

**/nt60** - запись программного кода в загрузочные секторы для обеспечения загрузки файла bootmgr - диспетчера загрузки Windows Vista/Server 2008 и более поздних ОС семейства Windows.

**/force** - принудительное отключение используемых другими программами томов дисков для обеспечения монопольного доступа утилиты bootsect.exe

**/mbr** - изменение программного кода главной загрузочной записи (MBR - Master Boot Record) без изменения таблицы разделов диска. При использовании с параметром /nt52, MBR будет совместима с предшествующими Windows Vista версиями. При использовании с параметром /nt60 - MBR будет совместима с операционными системами Windows Vista и более поздними.

Пропишем **bootsect /nt60 /mbr C:** в командной строке и нажмем Enter. Должно появиться сообщение, что bootcode успешно обновлен.

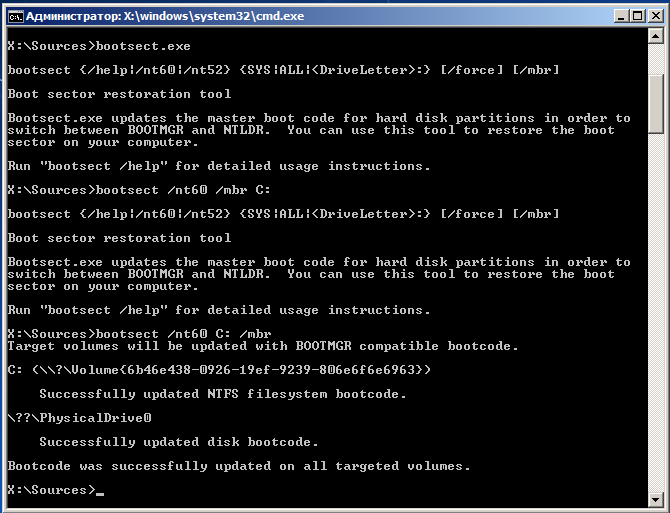


Рис. 24. Успешное обновление кода загрузчика.

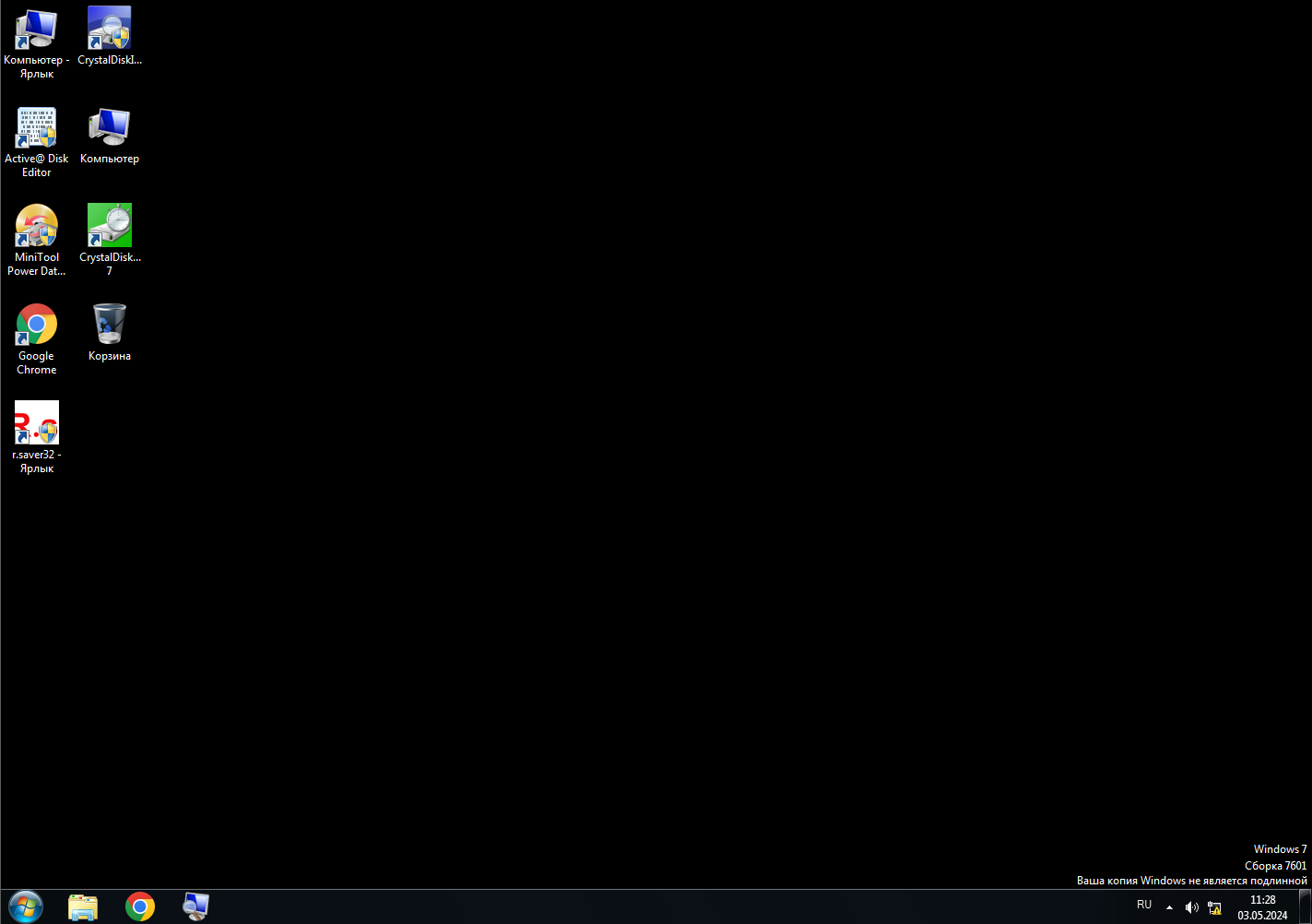


Рис. 25. Успешная загрузка ОС после восстановления загрузчика.

**ЗАДАНИЕ 9.** Основы работы с командной строкой Windows.

Для автоматизации процесса обработки вводимых команд возможно создавать простейшие скрипты на языке оболочки командной строки. попробуем автоматизировать процесс создания дерева папок. Создадим папку firstcat, в которой будет папка secondcat, которая в свою очередь будет содержать папку thirdcat.

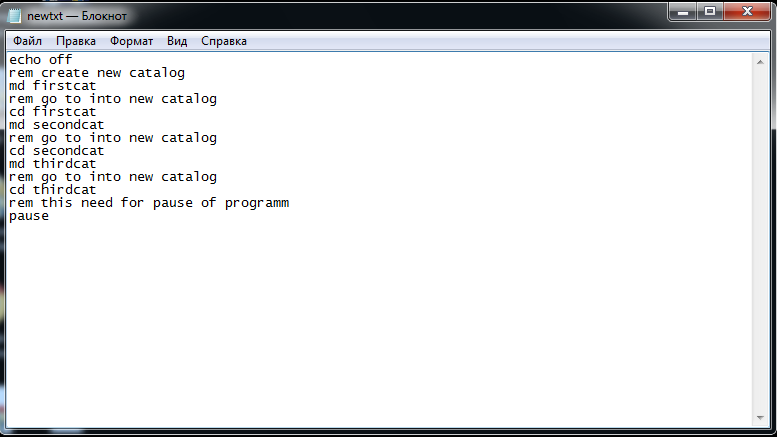


Рис. 26. Создание скрипта для создания дерева папок.

После написания скрипта закрываем текстовый редактор и меняем расширение файла на .bat, чтобы он стал доступен для исполнения и чтения командной строкой.



Рис. 27. Готовый к исполнению скрипт.

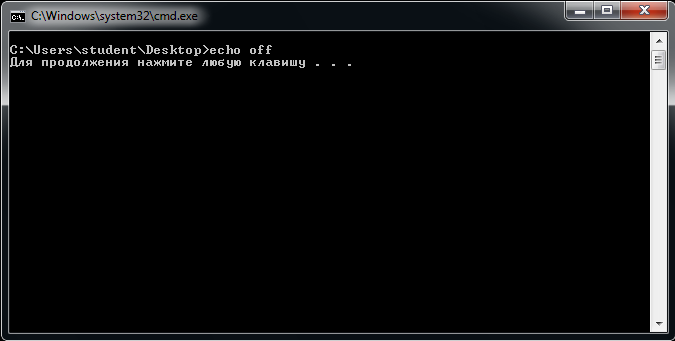


Рис. 28. Успешный запуск скрипта.

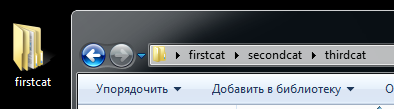


Рис. 29. Созданное дерево папок.

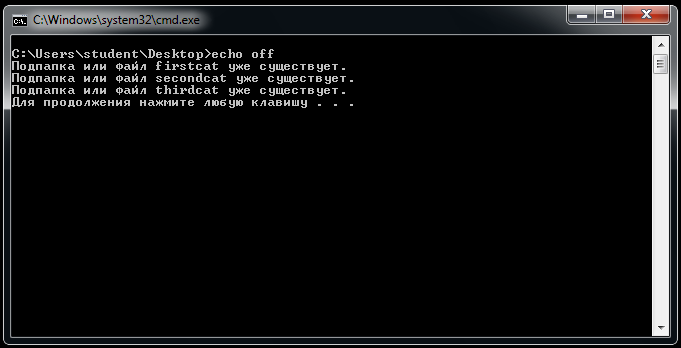


Рис. 30. Неудачная попытка создать дерево папок при повторном запуске.

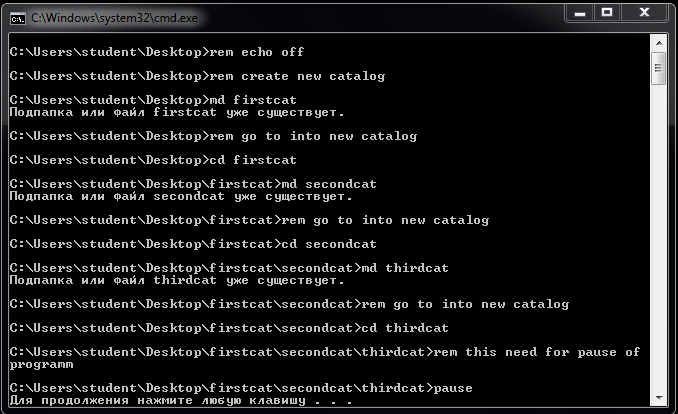


Рис. 31. Вывод всех команд скрипта в консоль при комментировании строки echo off.

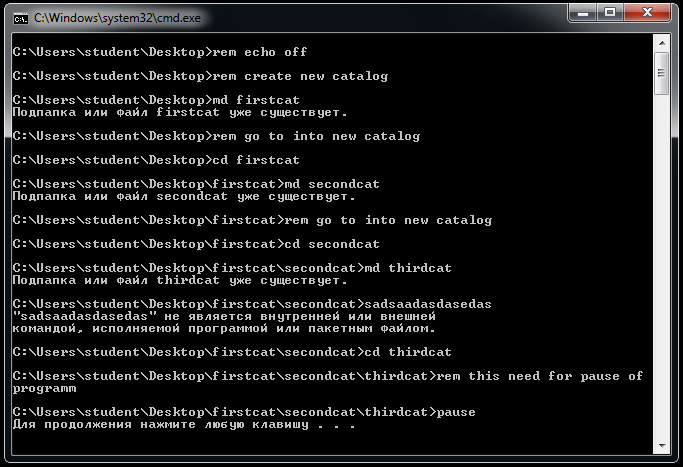


Рис. 32. Замена строчки на случайные латинские символы даёт ошибку исполнения этой строчки.

Если же убрать из скрипта pause, то после того, как консоль исполнит все строки, она сразу закрывается.

Напишем программу в новом файле .bat, которая удаляет папки. Для того, чтобы удалить все папки, необходимо двигаться в обратном порядке. Вначале необходимо зайти в первый каталог, из него перейти во второй каталог, после чего удалить третий каталог. Далее вернуться на уровень выше в первый каталог и удалить второй, после чего выйти из первого и удалить первый.

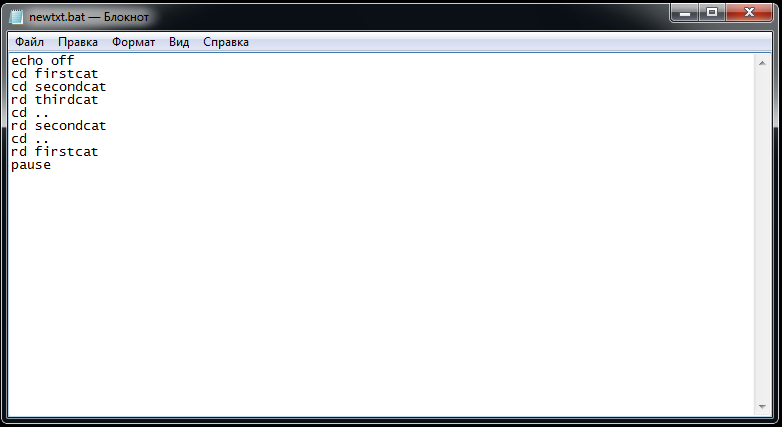


Рис. 33. Скрипт для удаления папок.

Если же попробовать удалить папки, а внутри них файлы, то выдаст следующее.

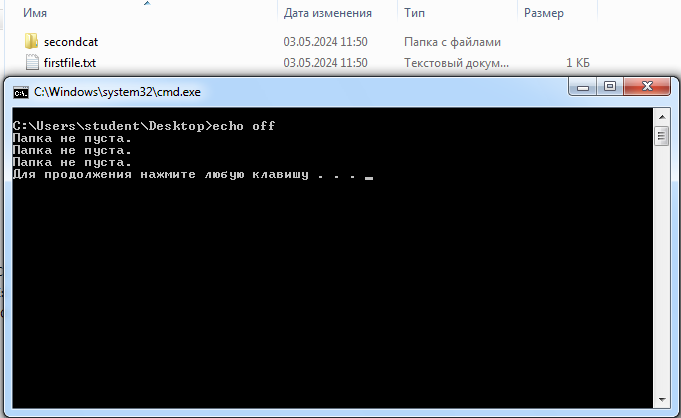


Рис. 34. Невозможно удалить непустые каталоги.

**Индивидуальное задание**

1. Создайте скрипт, который строит дерево каталогов, верхнем будет папка с вашей фамилией, в ней папки с именем и фамилией.

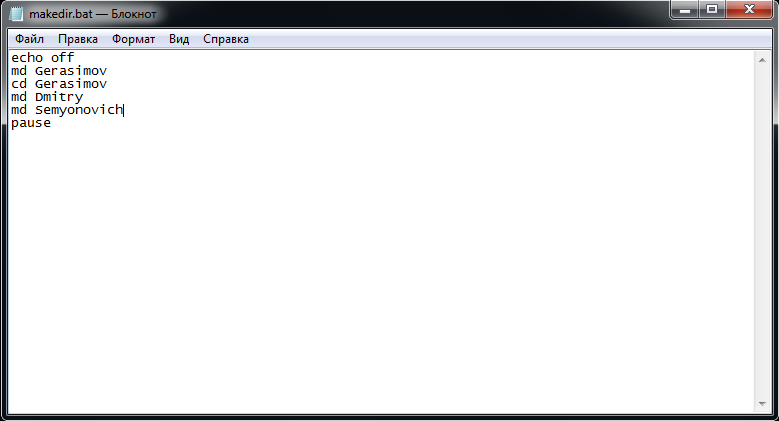


Рис. 35. Написанный скрипт для первого индивидуального задания.

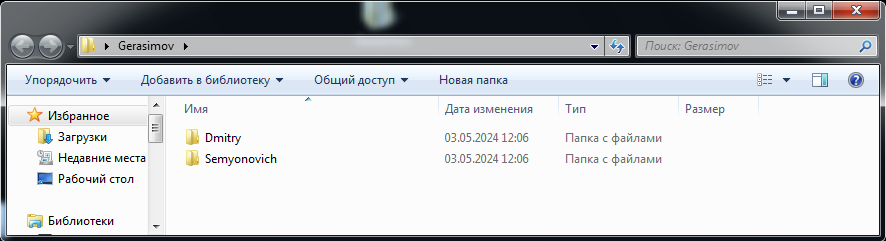


Рис. 36. Созданные с помощью скрипта каталоги.

2. Создайте скрипт, который создавал в папке с вашей фамилией файл, название которого это дата вашего рождения (например, 20121999), а в папке с вашим отчеством – файл, имя которого – номер компьютера, за которым вы сидите.

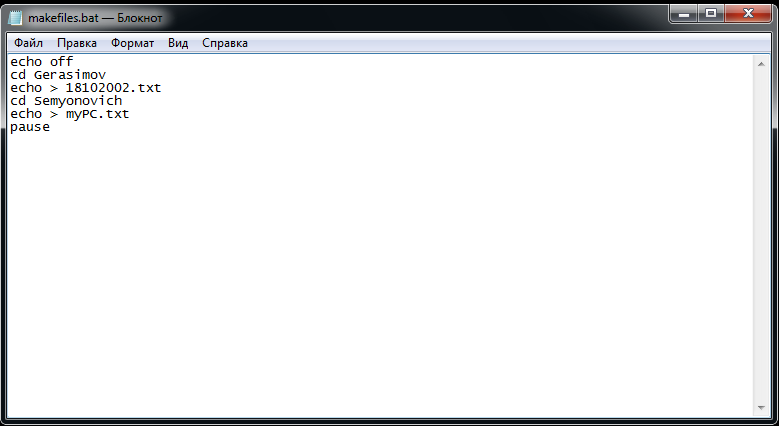


Рис. 37. Написанный скрипт для второго индивидуального задания.

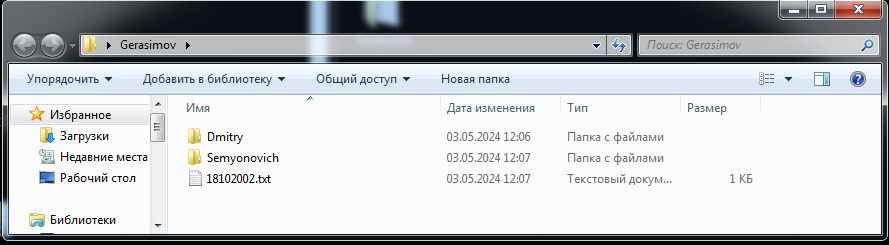


Рис. 38. Созданные с помощью скрипта каталоги и файл.

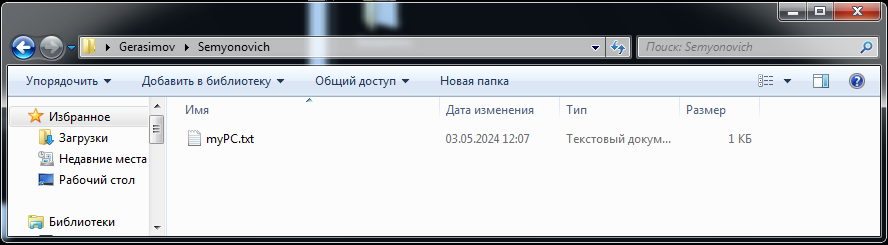


Рис. 39. Созданный файл внутри каталога Semyonovich.

3. Создайте скрипт, удаляющий файлы.

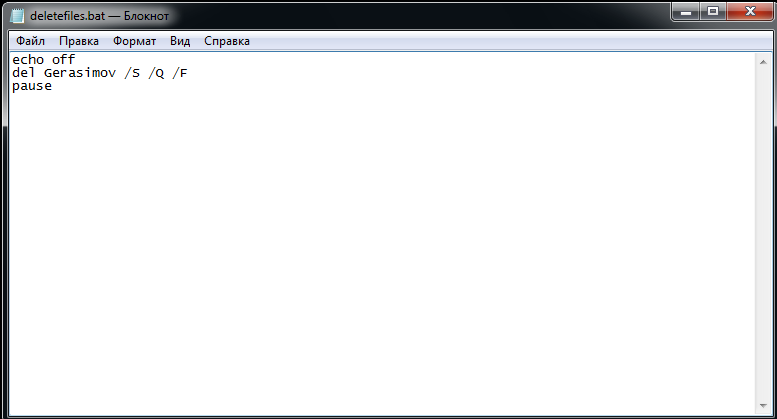


Рис. 40. Написанный скрипт для третьего индивидуального задания.



Рис. 41. Исполнение скрипта.

4. Создайте скрипт, удаляющий папки.

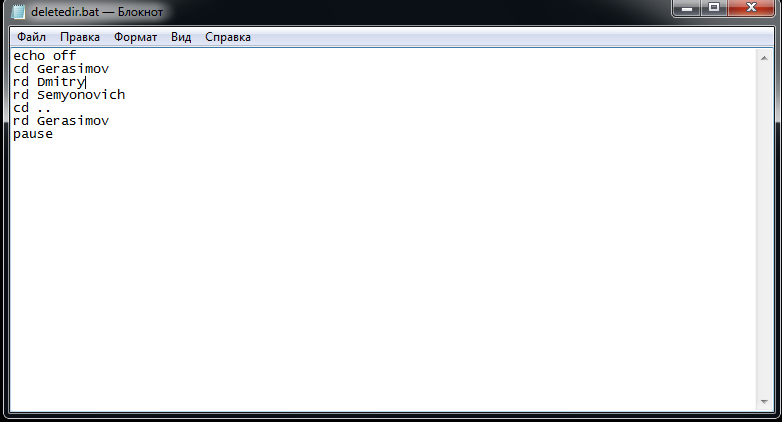


Рис. 42. Написанный скрипт для четвертого индивидуального задания.

5. Объедините эти скрипты в один так, чтобы после каждого действия (создание папок, создание файлов, удаление файлов, удаление папок) пользователю предлагалось нажать любую клавишу.

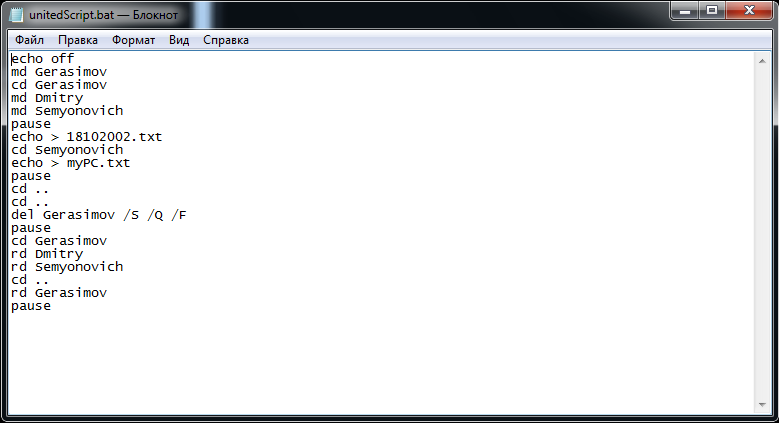


Рис. 43. Написанный скрипт для пятого индивидуального задания.

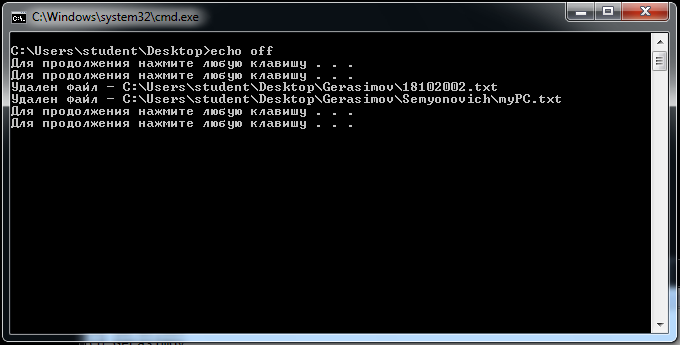


Рис. 44. Исполнение скрипта.

**Вывод**

По ходу выполнения лабораторной работы удалось овладеть навыками работы с консольной утилитой DiskPart для управления разделами жестких дисков, навыками редактирования секторов жестких дисков и логических разделов. Были изучены основы работы с командной строкой Windows и сделаны индивидуальные задания по написанию скриптов для создания и удаления каталогов и папок.