ГУАП

КАФЕДРА № 42

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Доцент |  |  |  | А.В. Аграновский |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ № 5 |
| **Разработка командлетов PowerShell**  Вариант 5 |
|  |
| по курсу: ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА ИНФОРМАЦИОННЫХ СИСТЕМ |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4128 |  |  |  | В.А. Воробьев |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2023

**СОДЕРЖАНИЕ**

[1 Цель работы 3](#_Toc129032861)

[2 Выполнение упражнений 4](#_Toc129032862)

[2.1 Упражнение 2.5. – Командлет Get-Service 4](#_Toc129032863)

[2.2 Упражнение 2.6. – Сортировка служб по статусу запущен или остановлен 5](#_Toc129032864)

[2.3 Упражнение 2.7. – Командлет Get-Process 6](#_Toc129032865)

[2.4 Упражнение 2.8. – Создание списков процессов 9](#_Toc129032866)

[2.5 Упражнение 2.9. – Подсчет количества объектов 12](#_Toc129032867)

[2.6 Упражнение 2.10. – Чтение содержимого текстовых файлов 16](#_Toc129032868)

[2.7 Упражнение 2.11. – Запись текста в файл 18](#_Toc129032869)

[3 Индивидуальное задание 20](#_Toc129032870)

[3.1 Изучение Stop-Process 20](#_Toc129032871)

[3.2 Изучение Restart-Service 24](#_Toc129032872)

[3.3 Изучение Format-Table 27](#_Toc129032873)

[4 Сведения о системе 31](#_Toc129032874)

[5 Вывод 32](#_Toc129032875)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 34](#_Toc129032876)

**1 Цель работы**

Изучение командлетов PowerShell. Усовершенствование навыков администрирования в ОС Windows.

**Задание:**

1. Изучить методические указания подраздела 2.3.
2. Выполнить все упражнения.
3. Результат выполнения зафиксировать на скриншотах.
4. Выполнить задания для самостоятельного изучения в соответствии с вариантом.
5. Сделать выводы о проделанной работе.

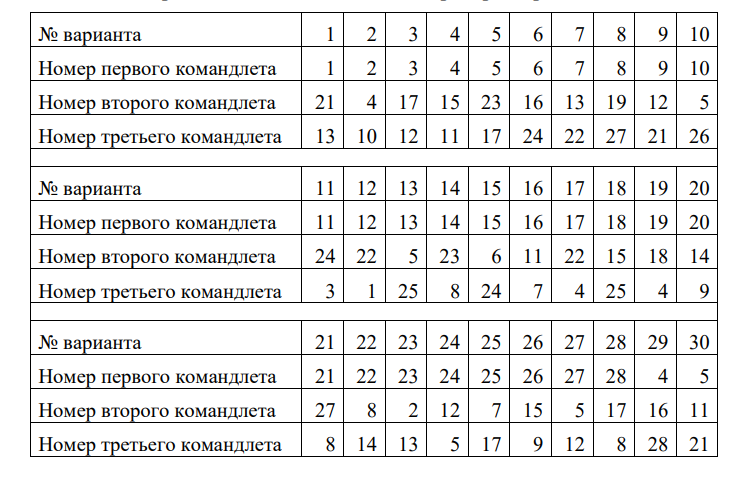


Рисунок 1 – Скриншот вариантов заданий

**2 Выполнение упражнений**

От нас требуется выполнить упражнения 2.5 – 2.11. При выполнении упражнений необходимо приложить скриншоты исполняемых команд.

**2.1 Упражнение 2.12. – Командлет Get-ChildItem**

Одним из псевдонимов Get-ChildItem является dir, так что далее ради удобства записи будем использовать его.

В 1 пункте упражнения введем свой каталог и расширение, ввиду отсутствия требуемых в задании.

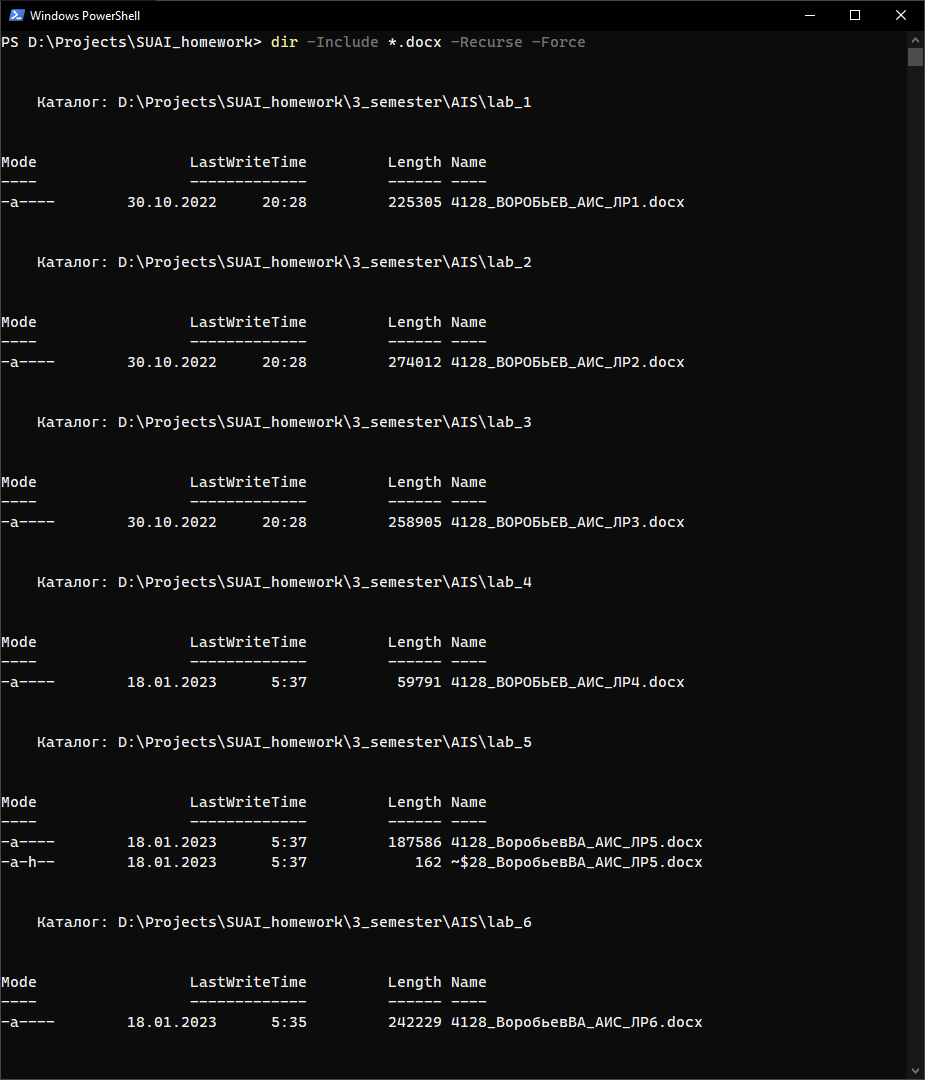


Рисунок 2 – Список служб

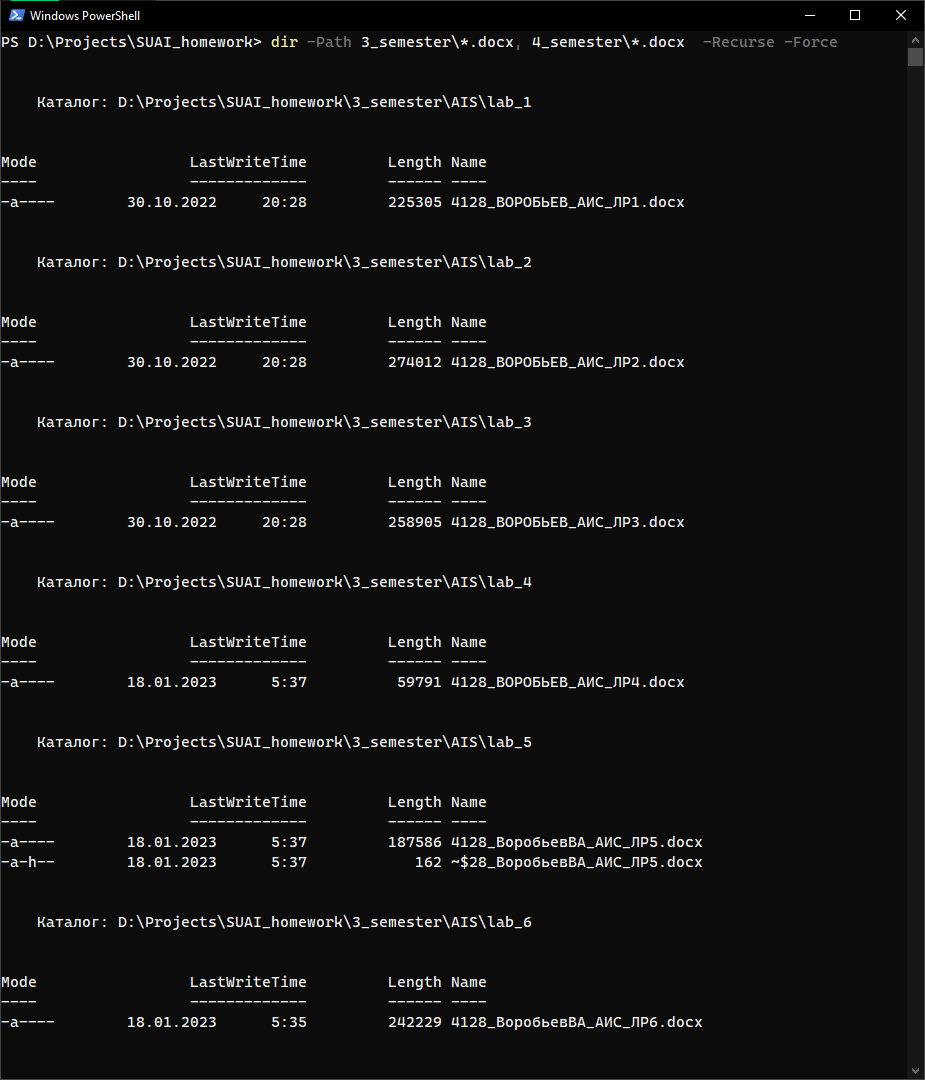
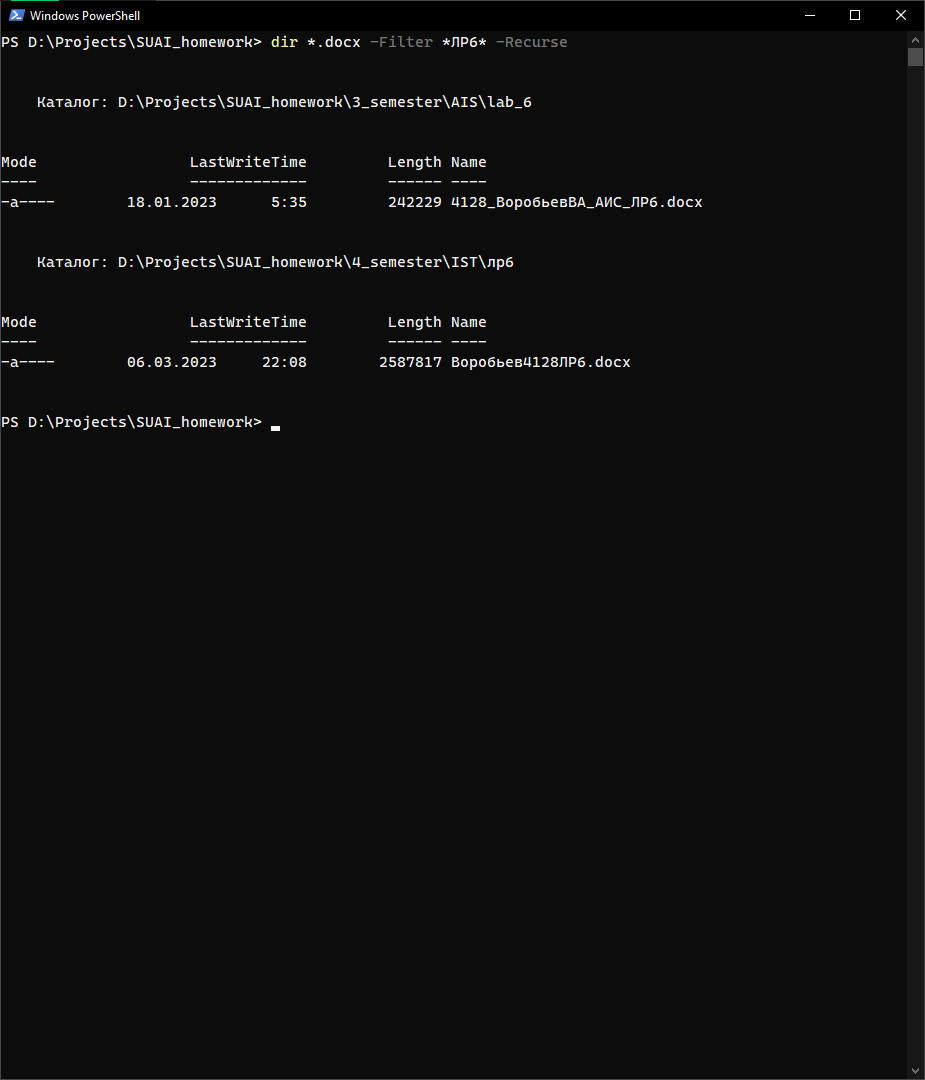
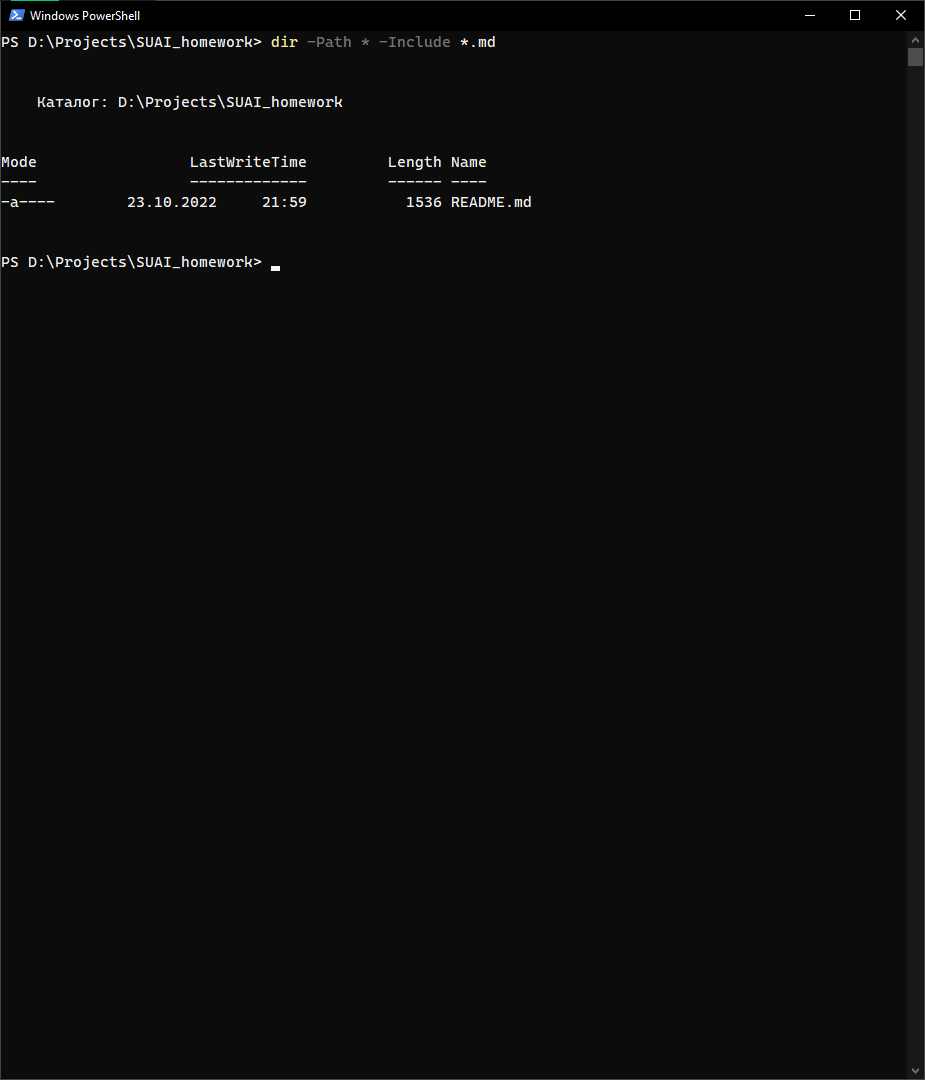
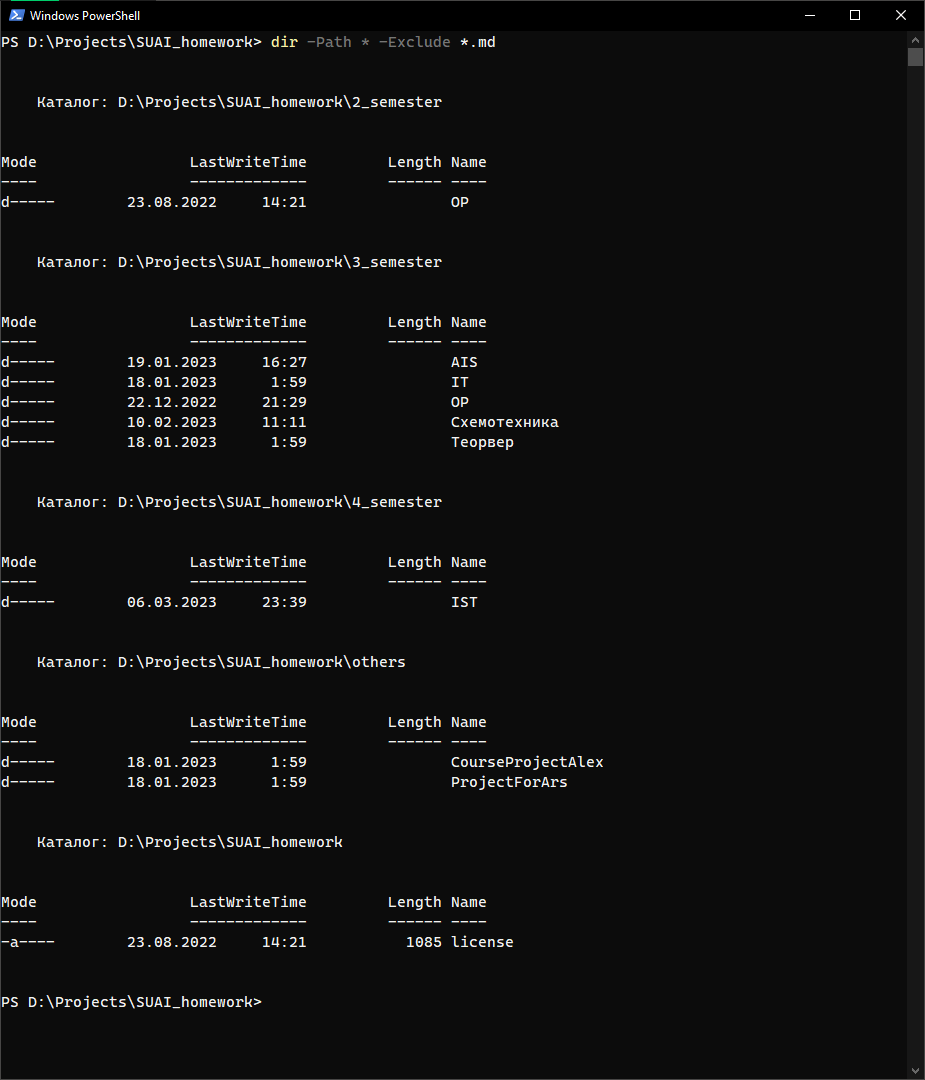


Рисунок 3 – Результат Get-Service







**2.2 Упражнение 2.13. – Командлет Get-Item**

В данном упражнении предлагается отсортировать службы по статусу используя механизм конвейеризации и командлет Sort-Object.

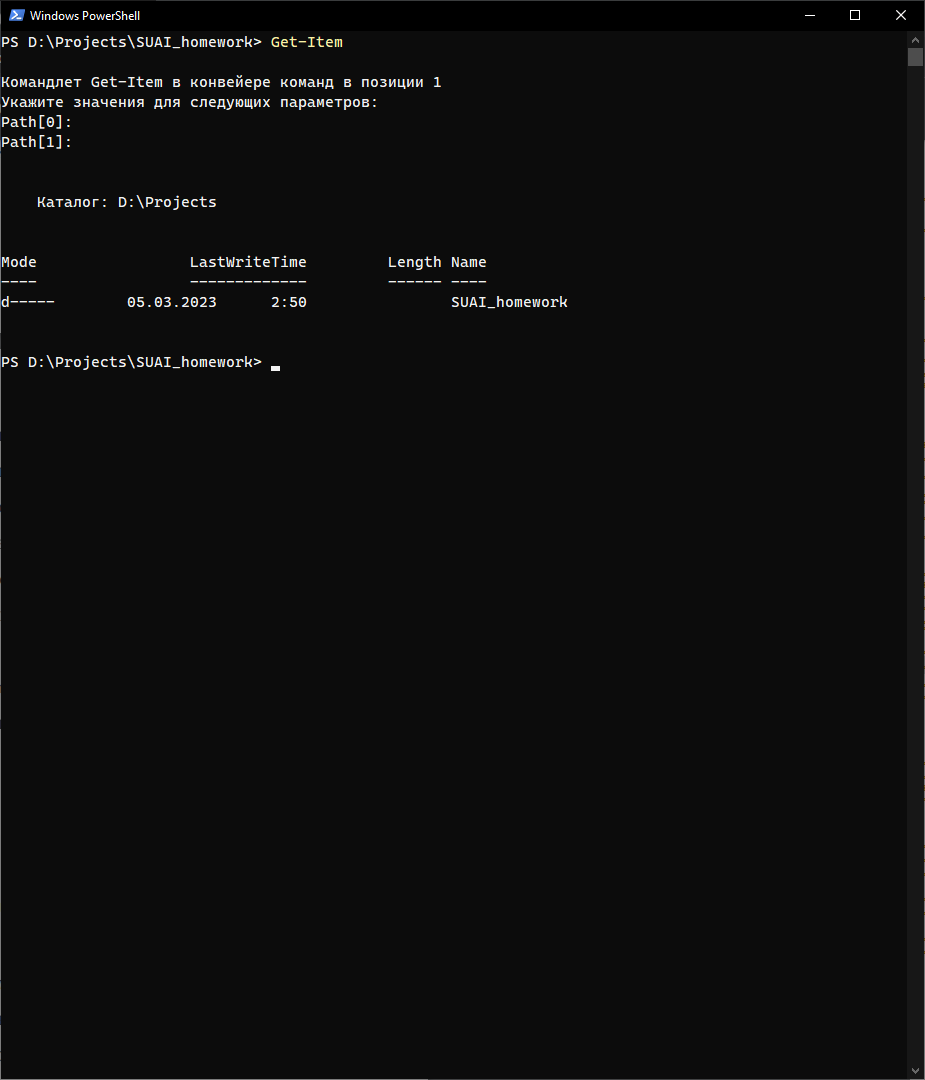
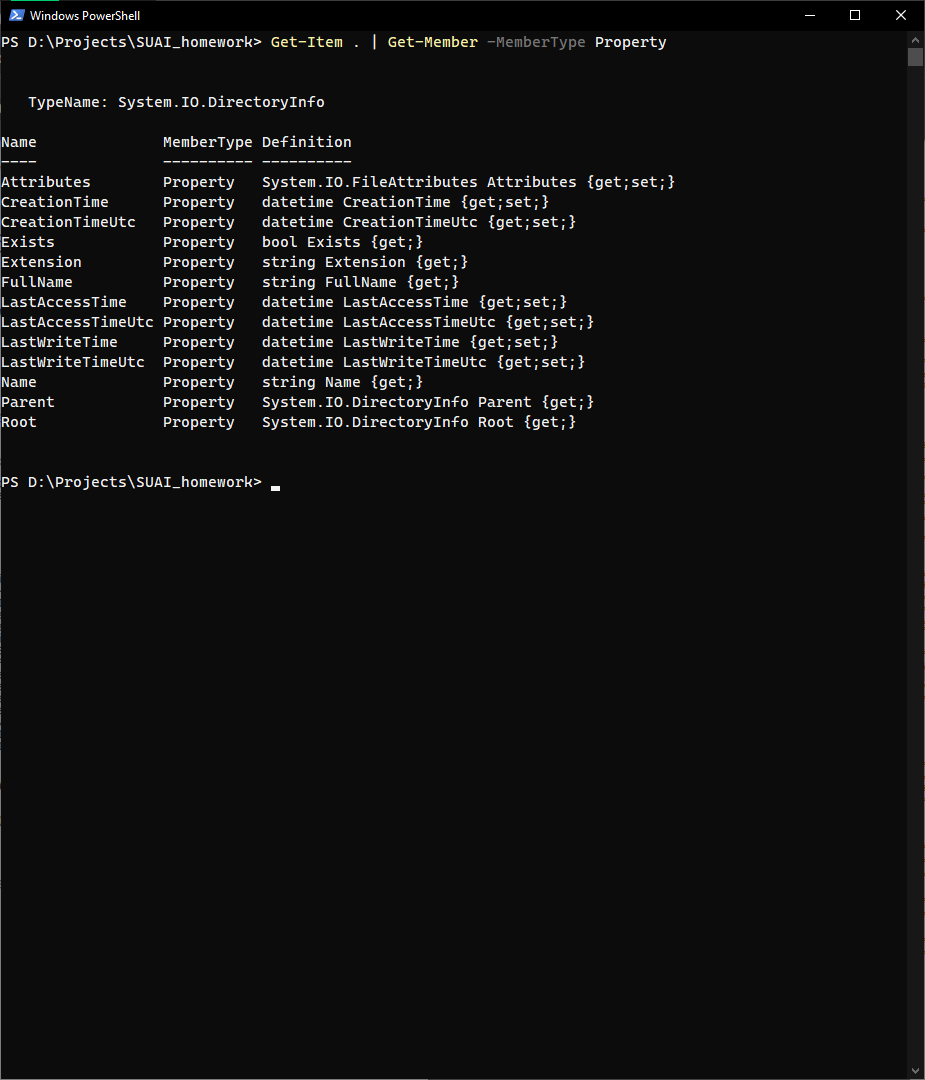
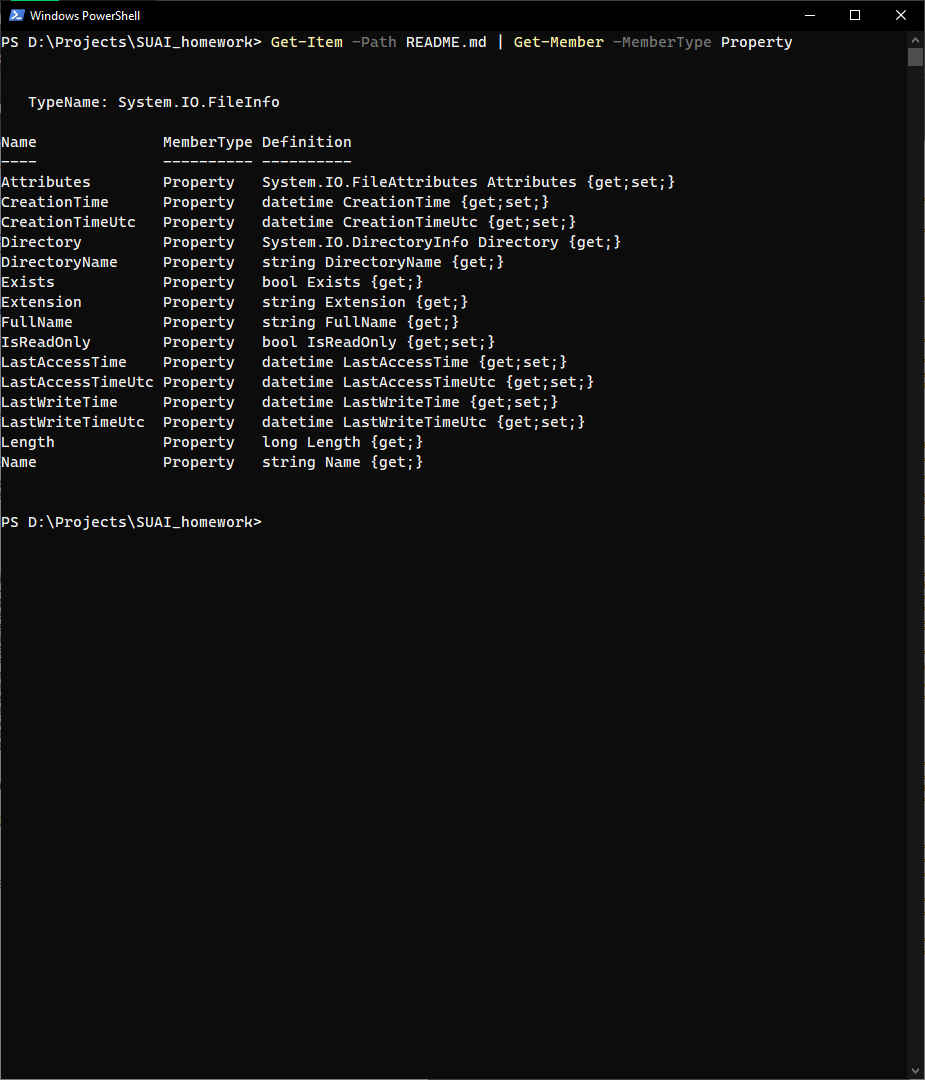


Рисунок 4 – Результат сортировки служб

****

****

**2.3 Упражнение 2.14. – Командлет New-Item**

В данном разделе предлагается изучить работу Get-Process в PowerShell. При анализе многопрограммной работы компьютера возникает задача определения, какие процессы активны в настоящее время и как используют ресурсы системы. Такую информацию можно получить, включив командлет Get-Process, который отражает данные о всех активных процессах.

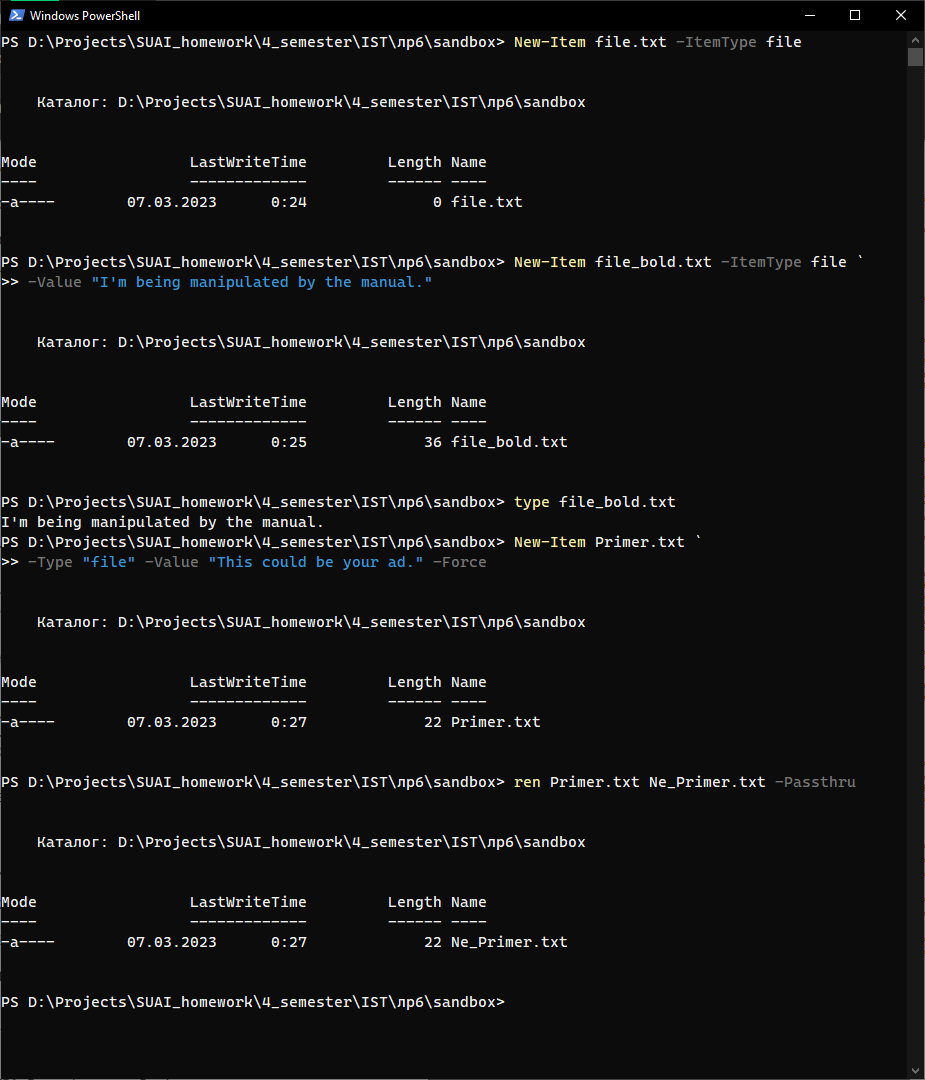


Рисунок 5 – Результат выполнения команды Get-Process

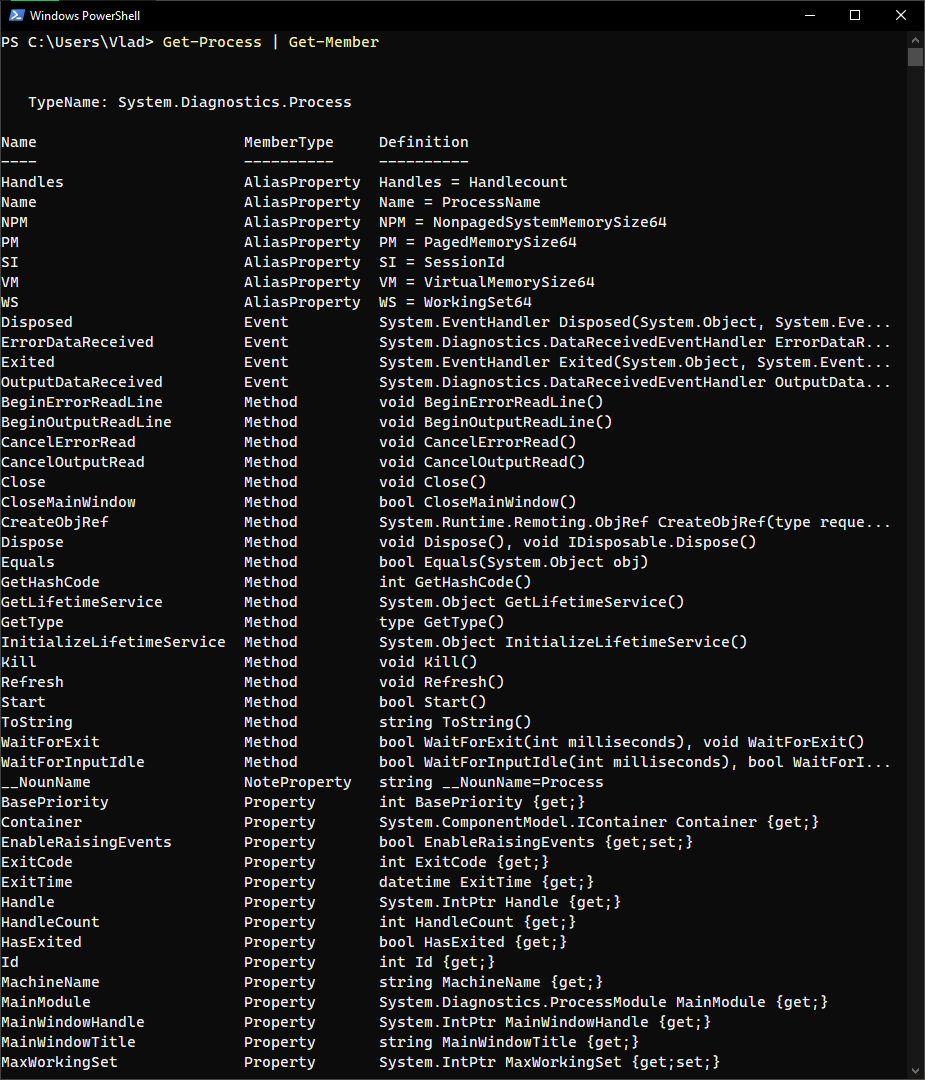


Рисунок 6 – Результат выполнении Get-Process в конвейере

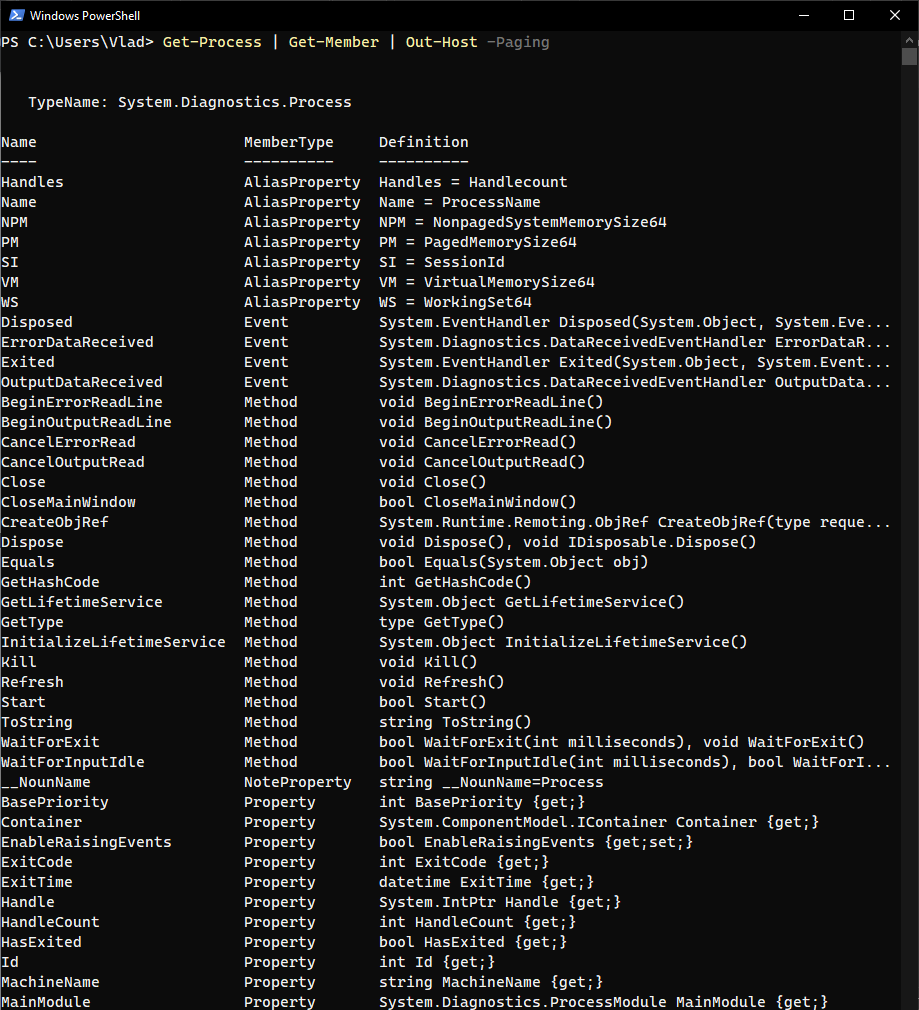


Рисунок 7 – Вывод процессов на одном экране

**2.4 Упражнение 2.15. – Работа с файлами средствами PowerShell.**

В данном упражнении предлагается ознакомиться с управление списков. Выполним упражнение приложив скриншоты.

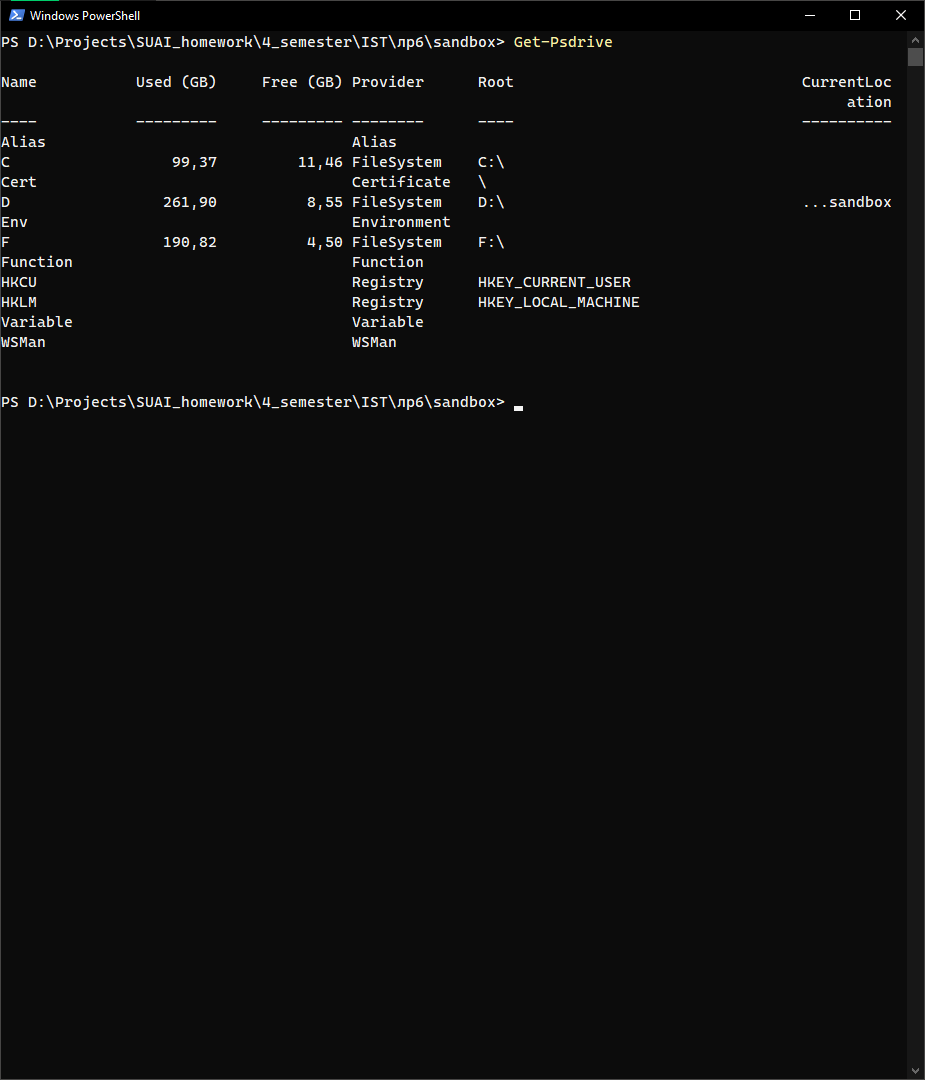
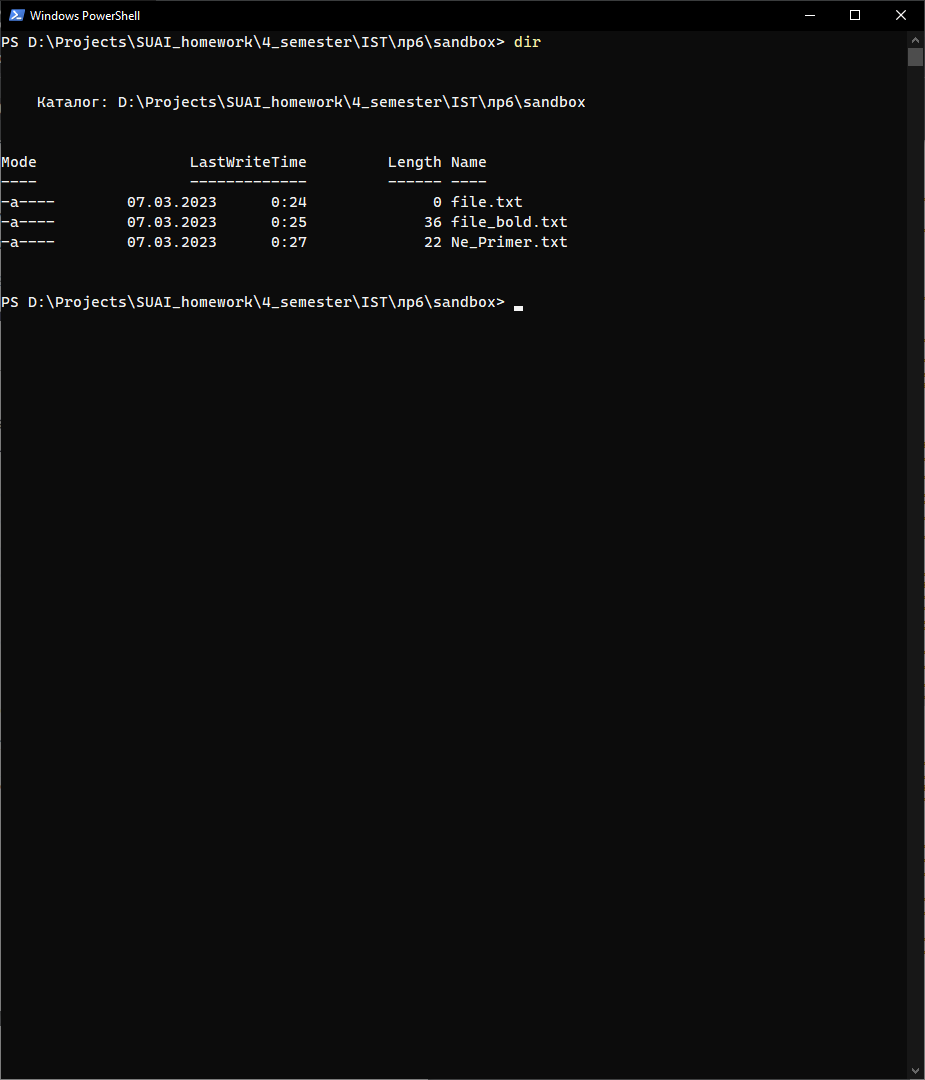
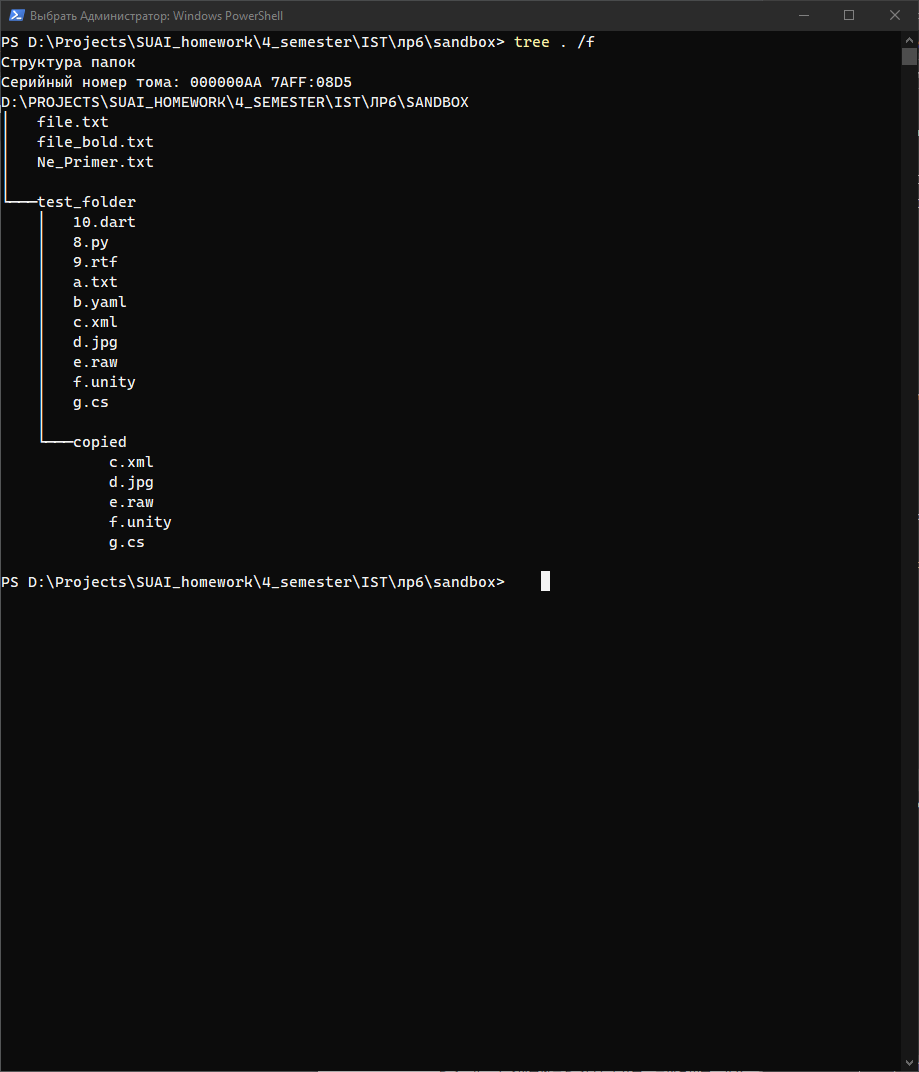
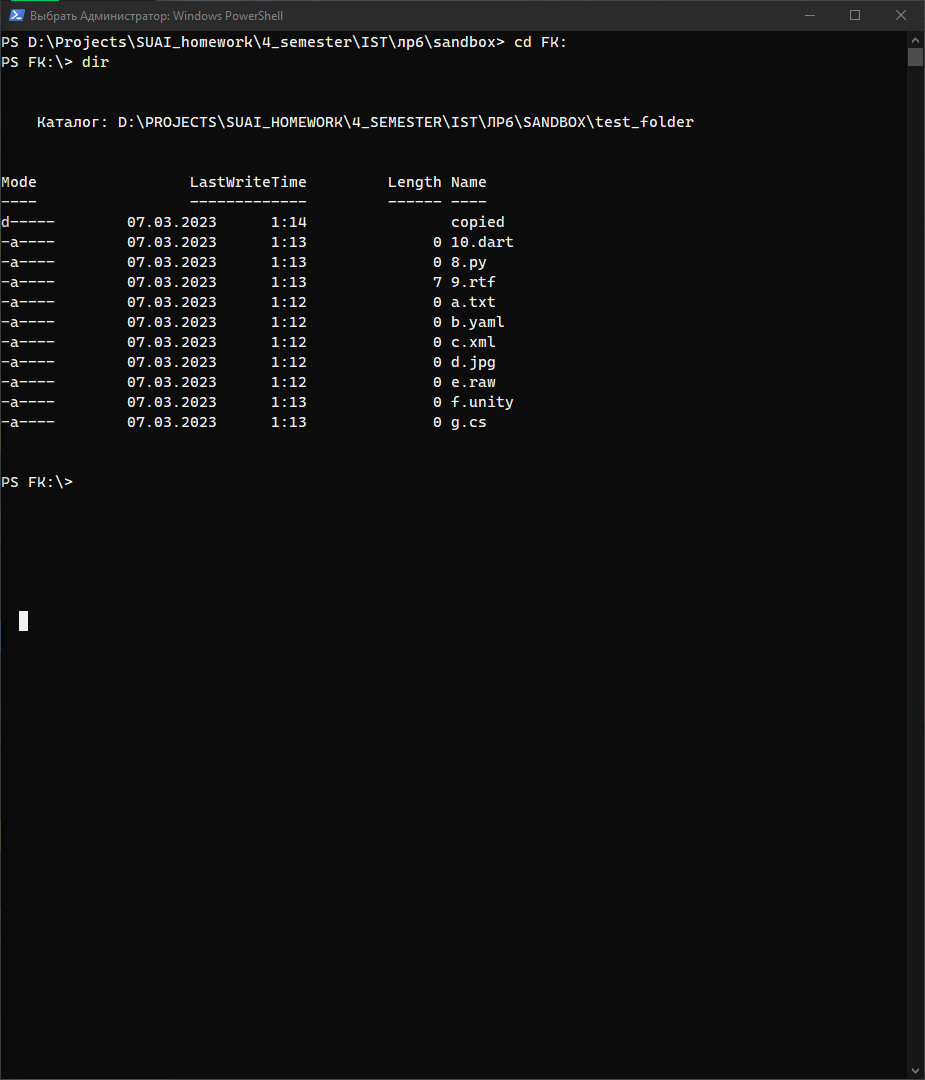
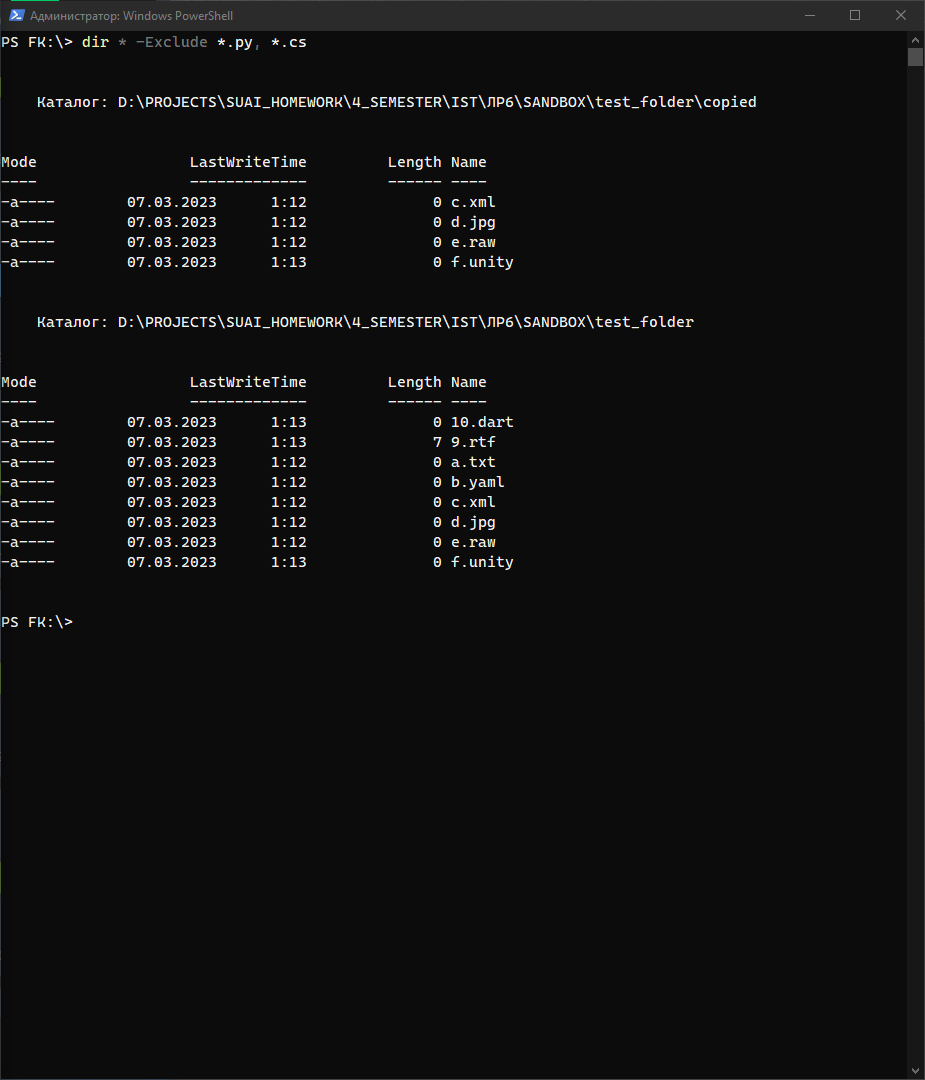


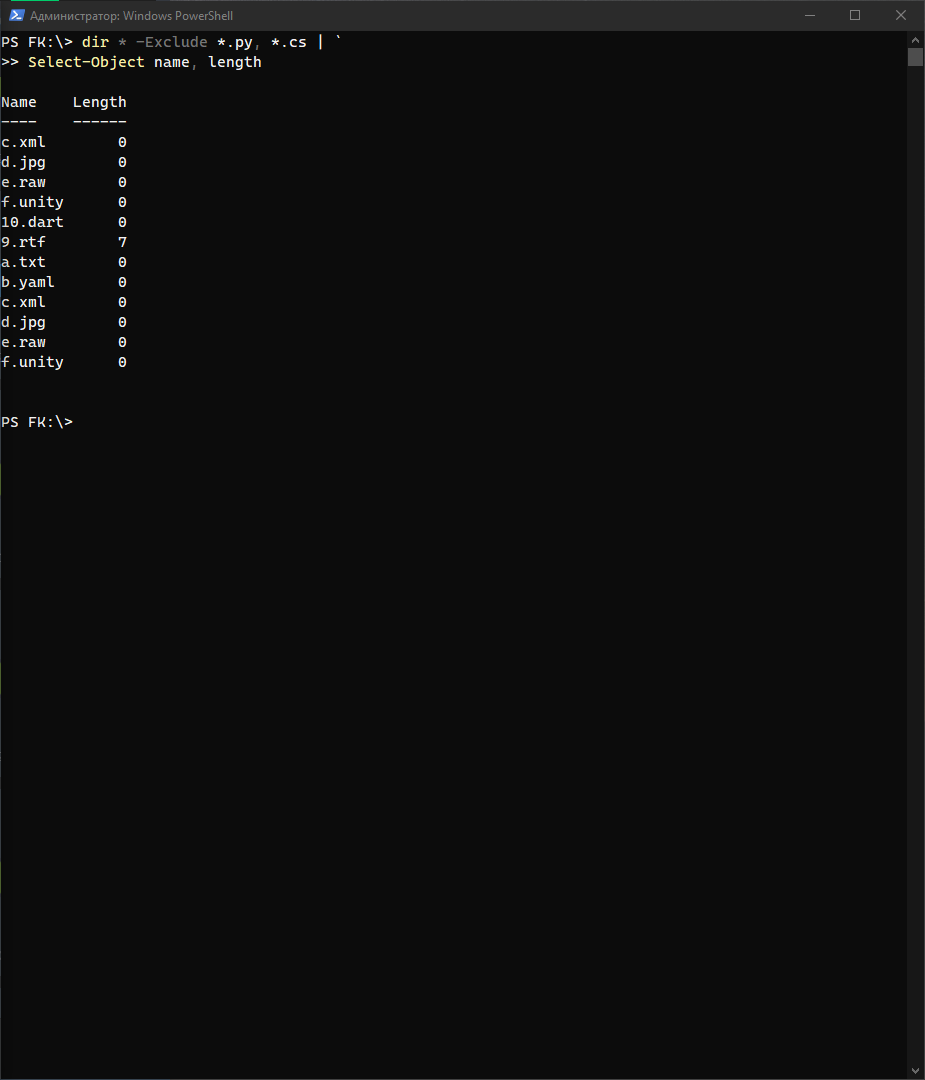
Рисунок 8 – Сортировка по времени использования CPU

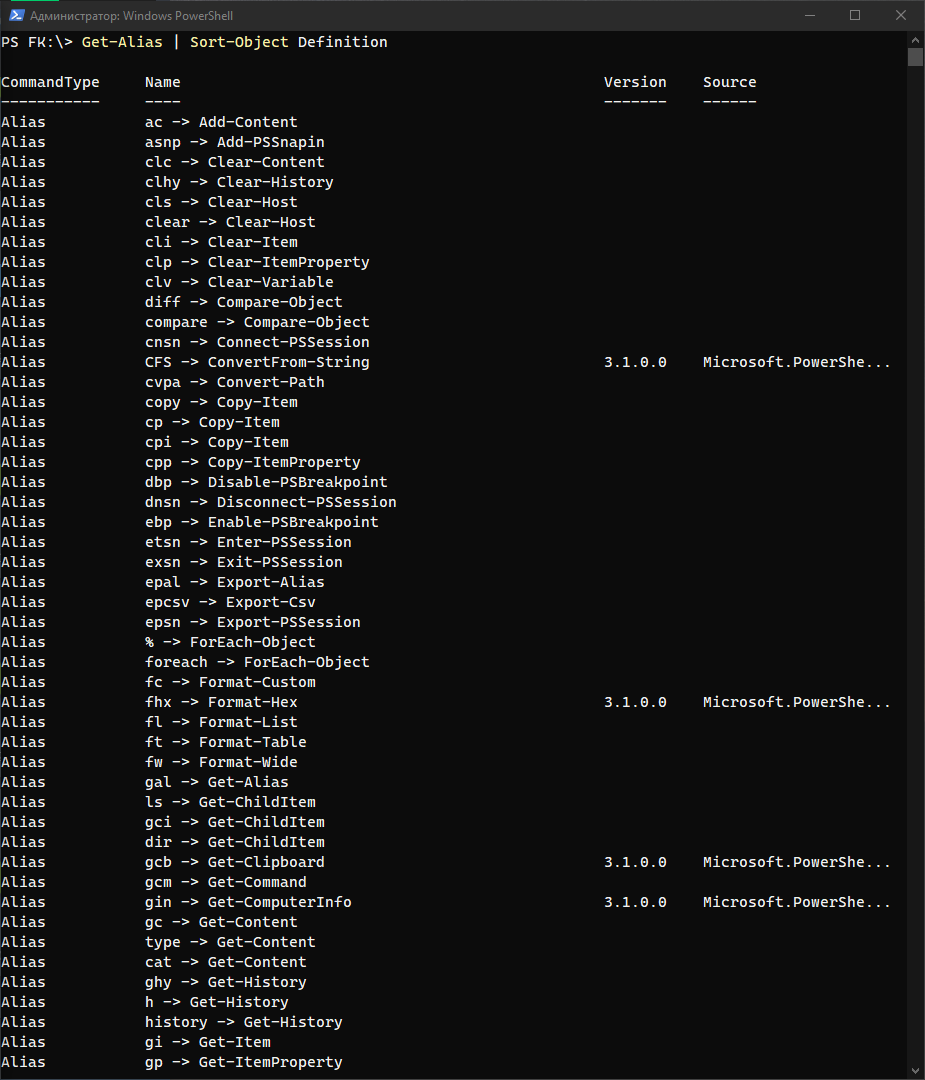


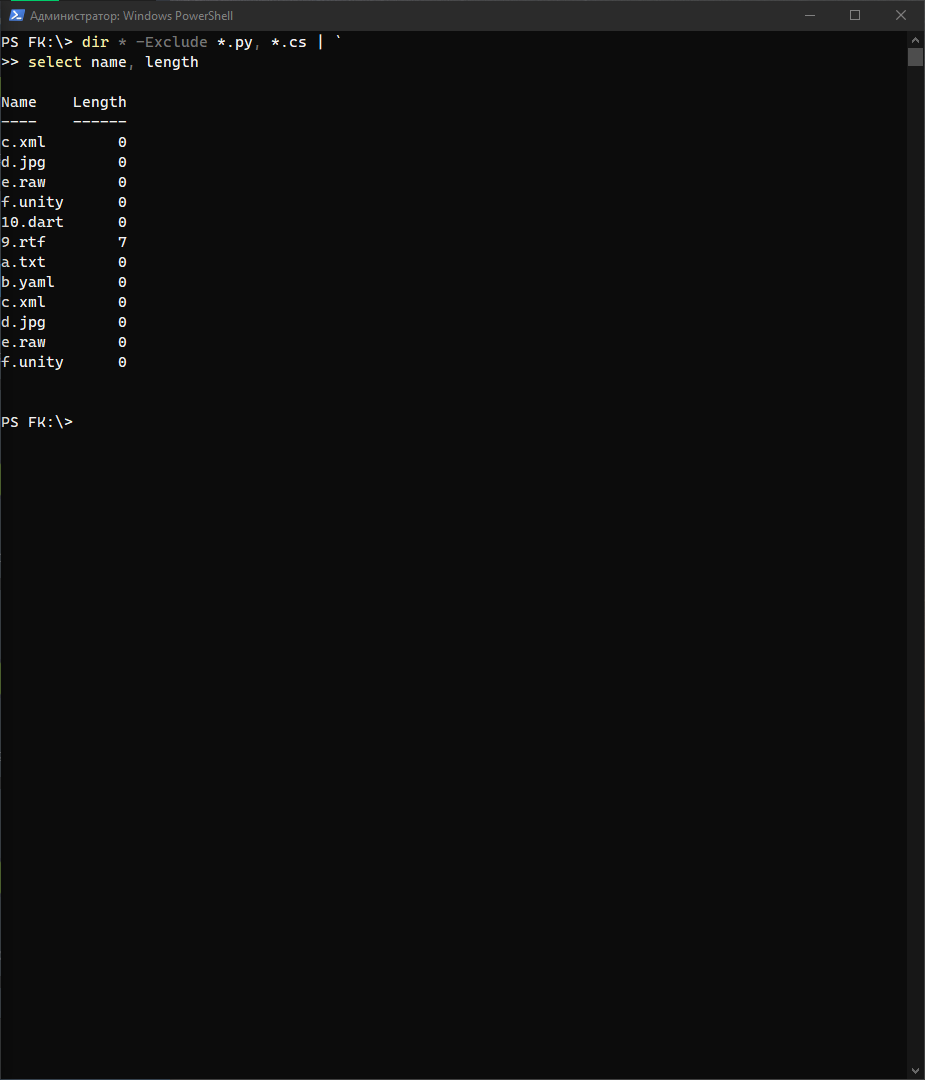


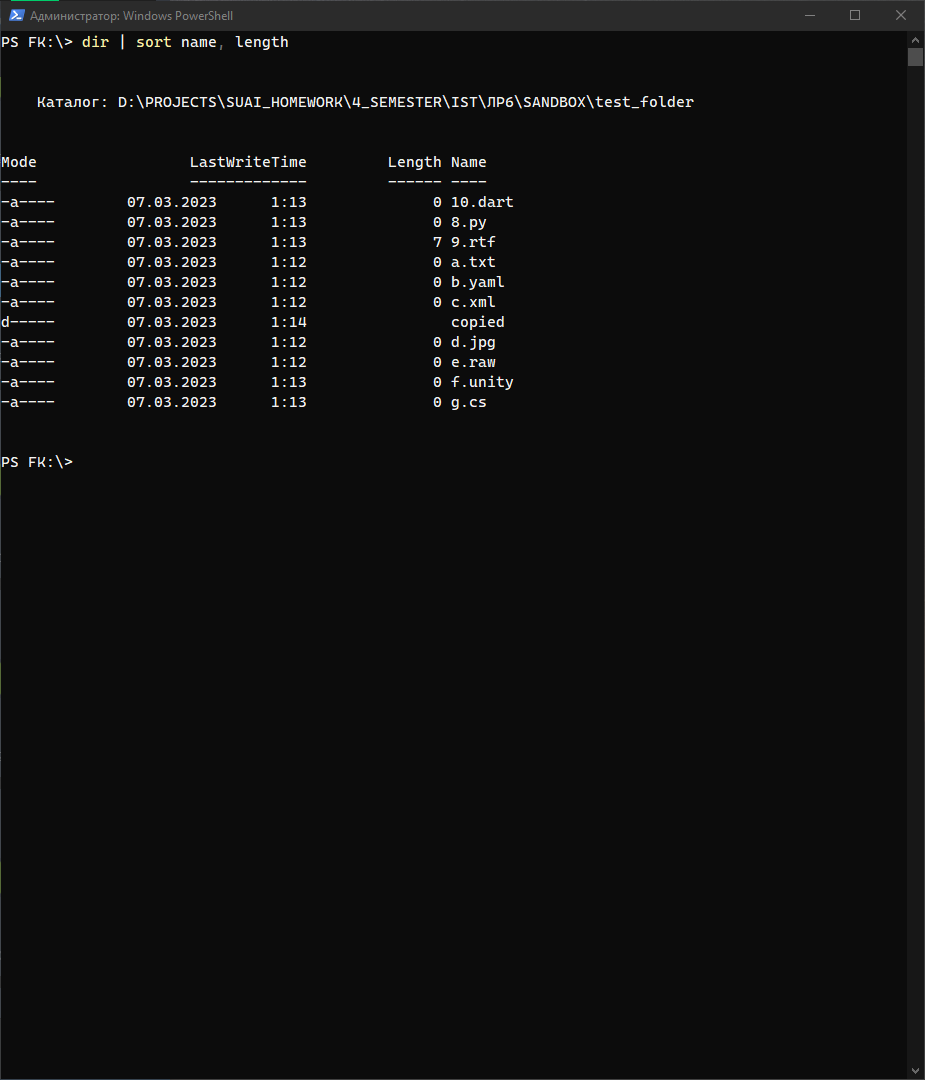


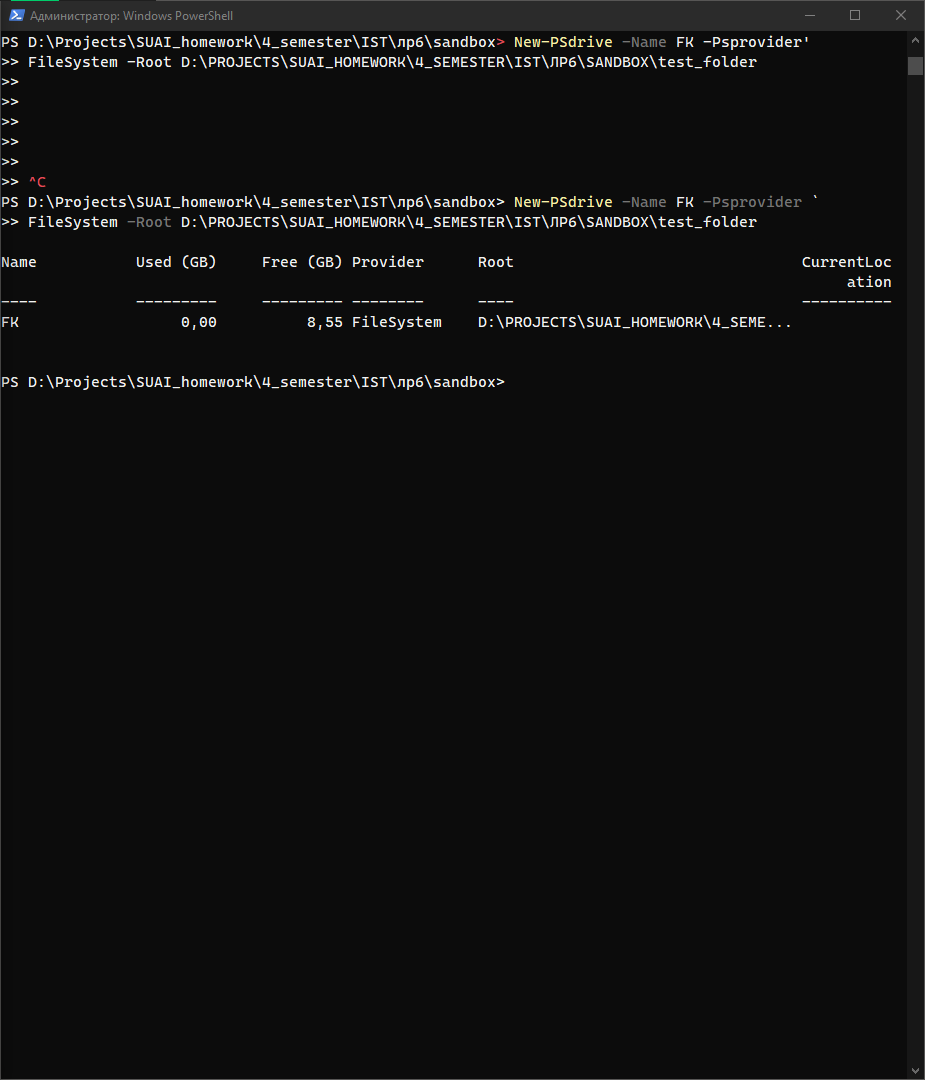












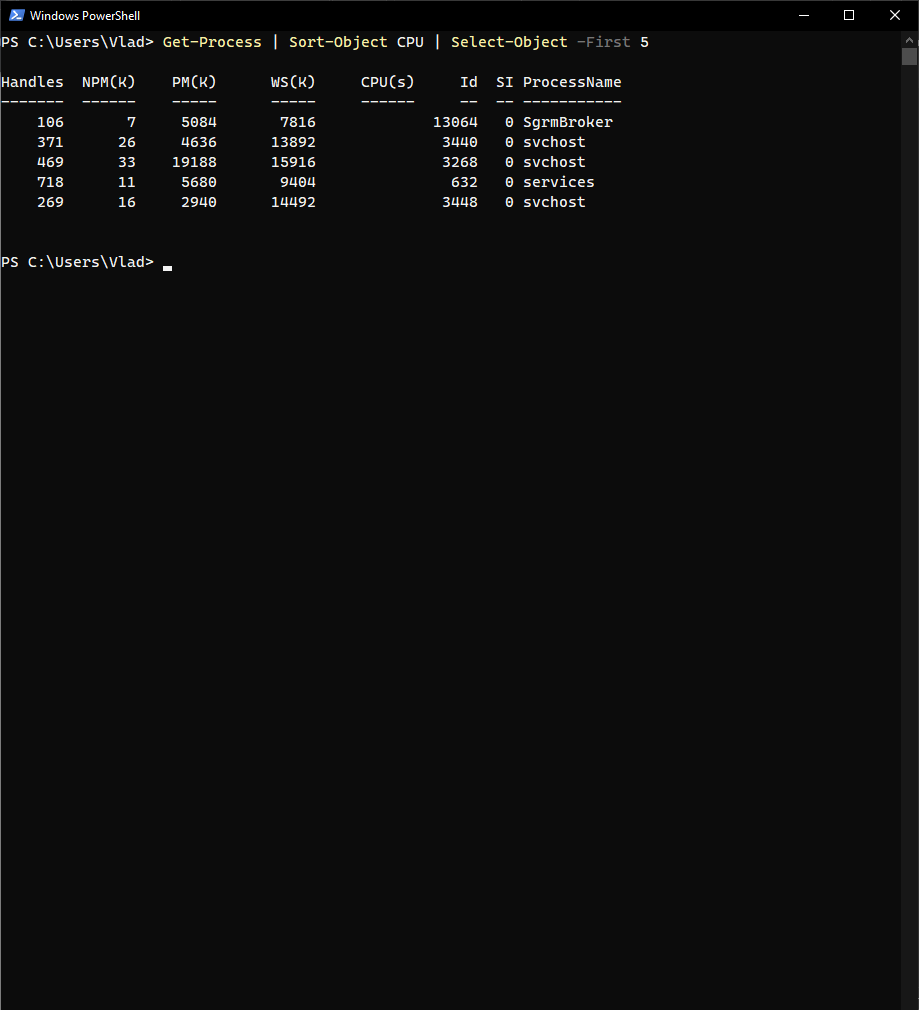


Рисунок 9 – Взятие 5 процессов по загруженности CPU

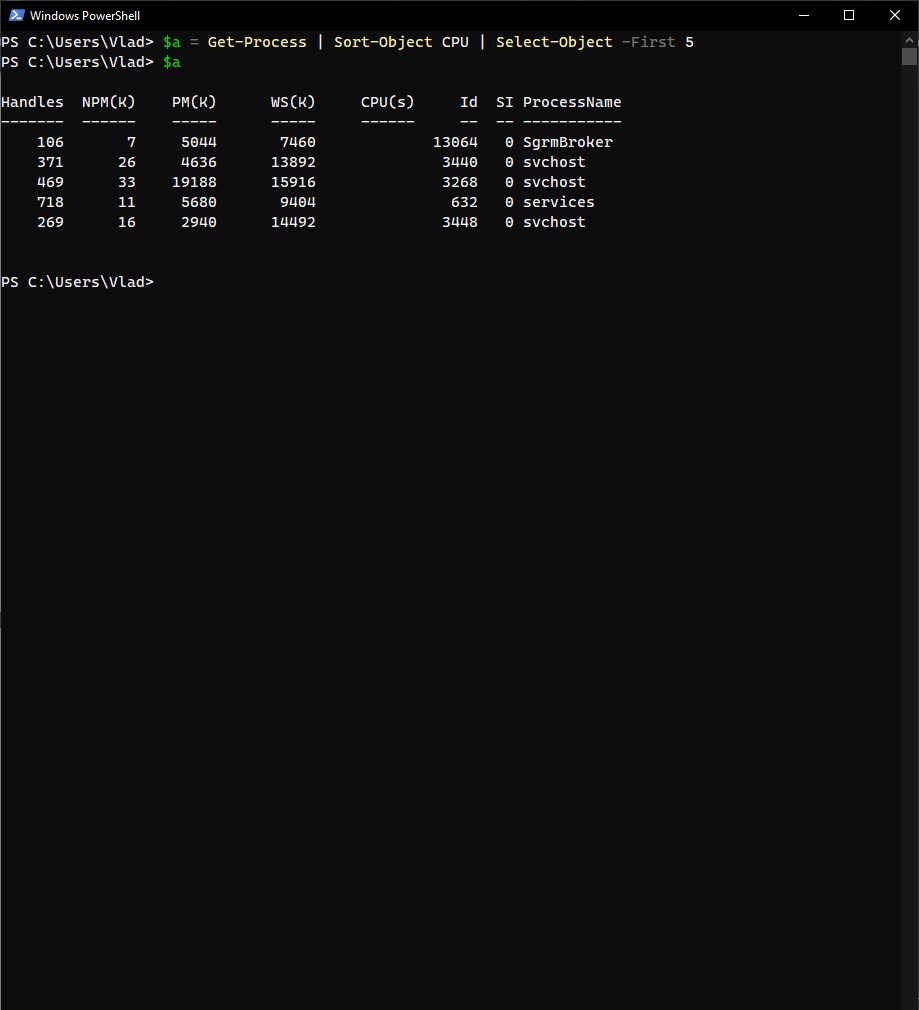


Рисунок 10 – Ввод и вывод переменной

**2.5 Упражнение 2.9. – Подсчет количества объектов**

В этом упражнении нам предлагается подсчитывать объекты с помощью командлета Measure-Object. Выполним упражнение и приложим скриншоты выполнения команд.

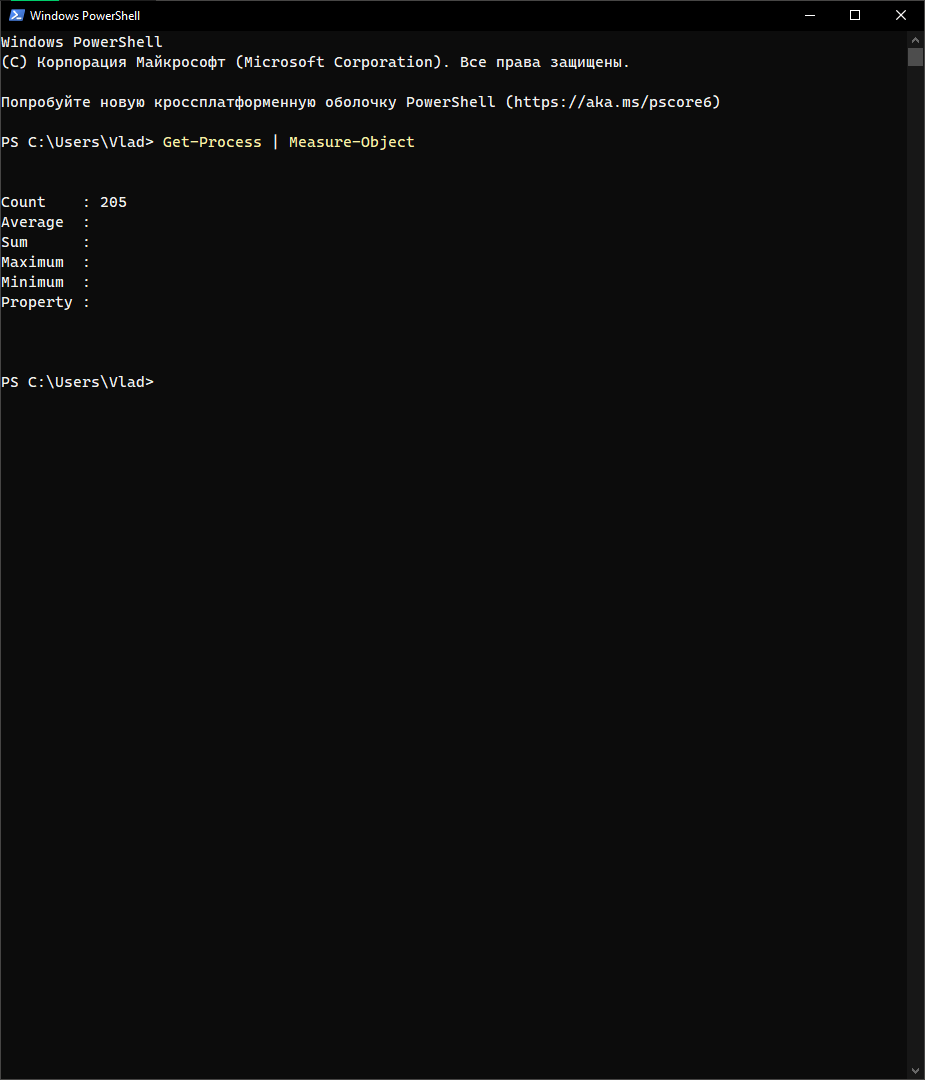


Рисунок 11 – Подсчет количества процессов

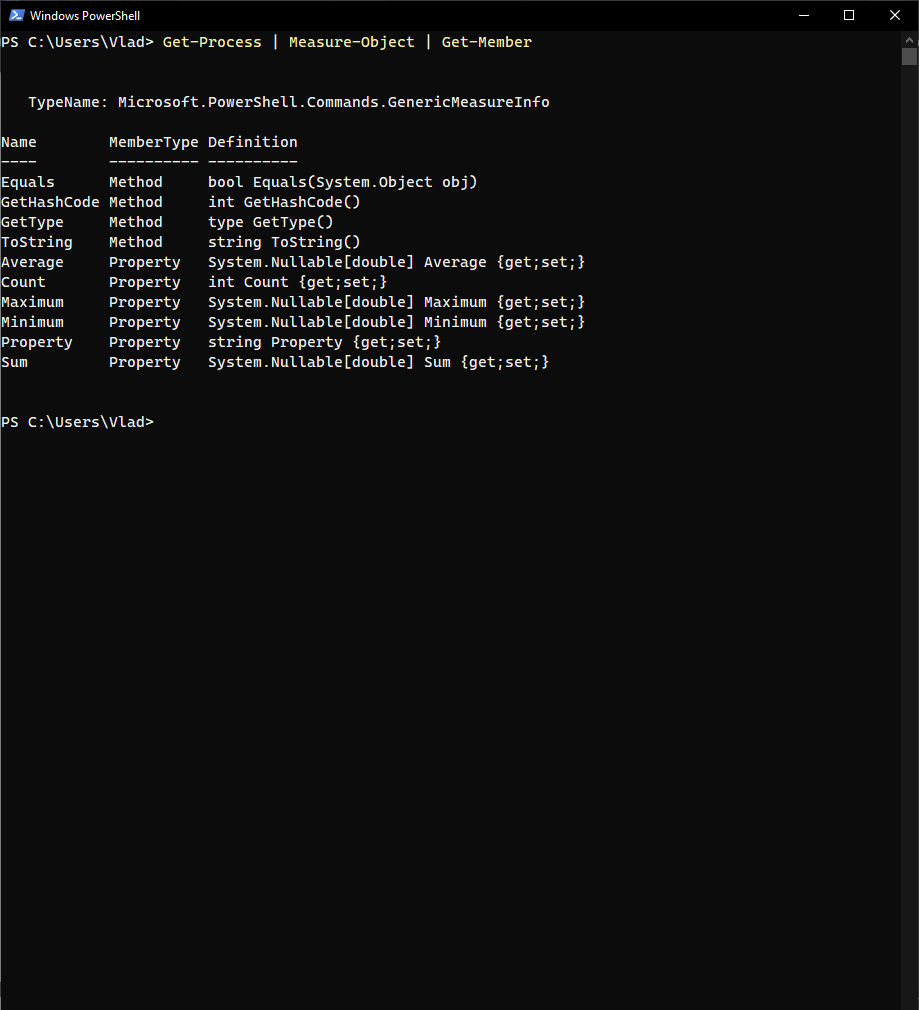


Рисунок 12 – Вывод членов объекта Measure-Object

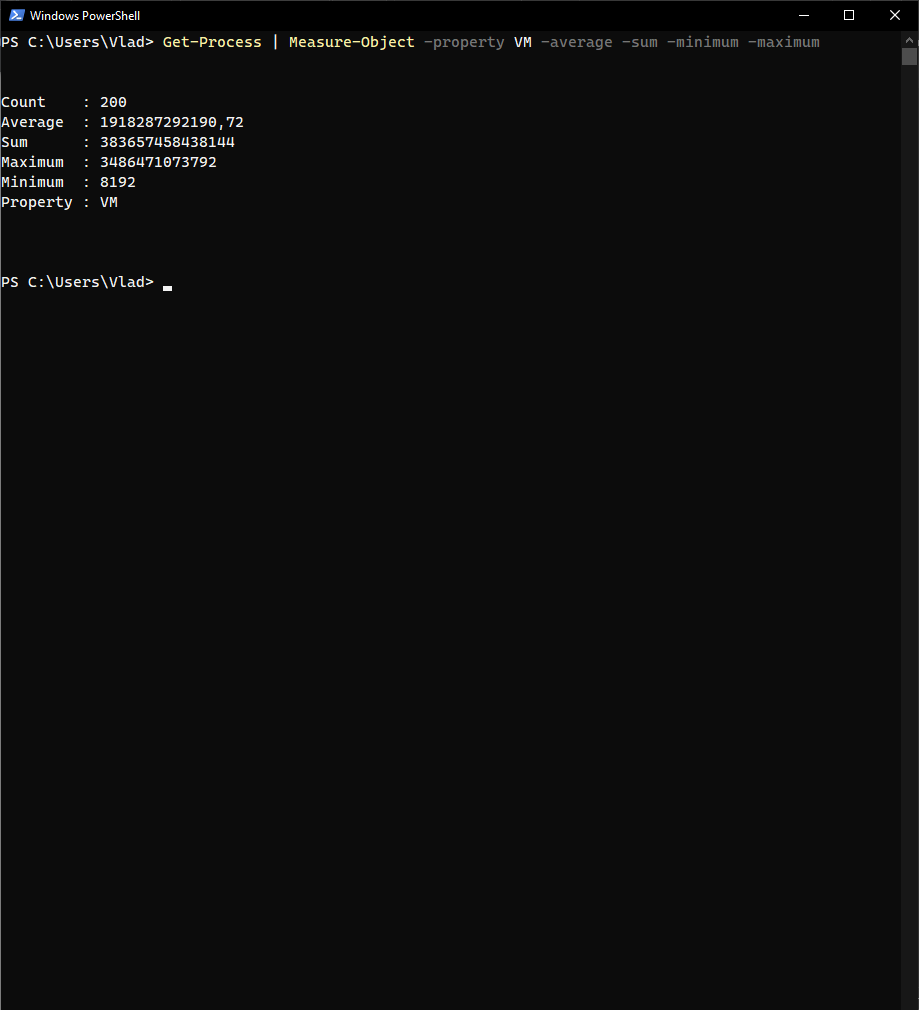


Рисунок 13 – Результат выполнения Measure-Object с ключами

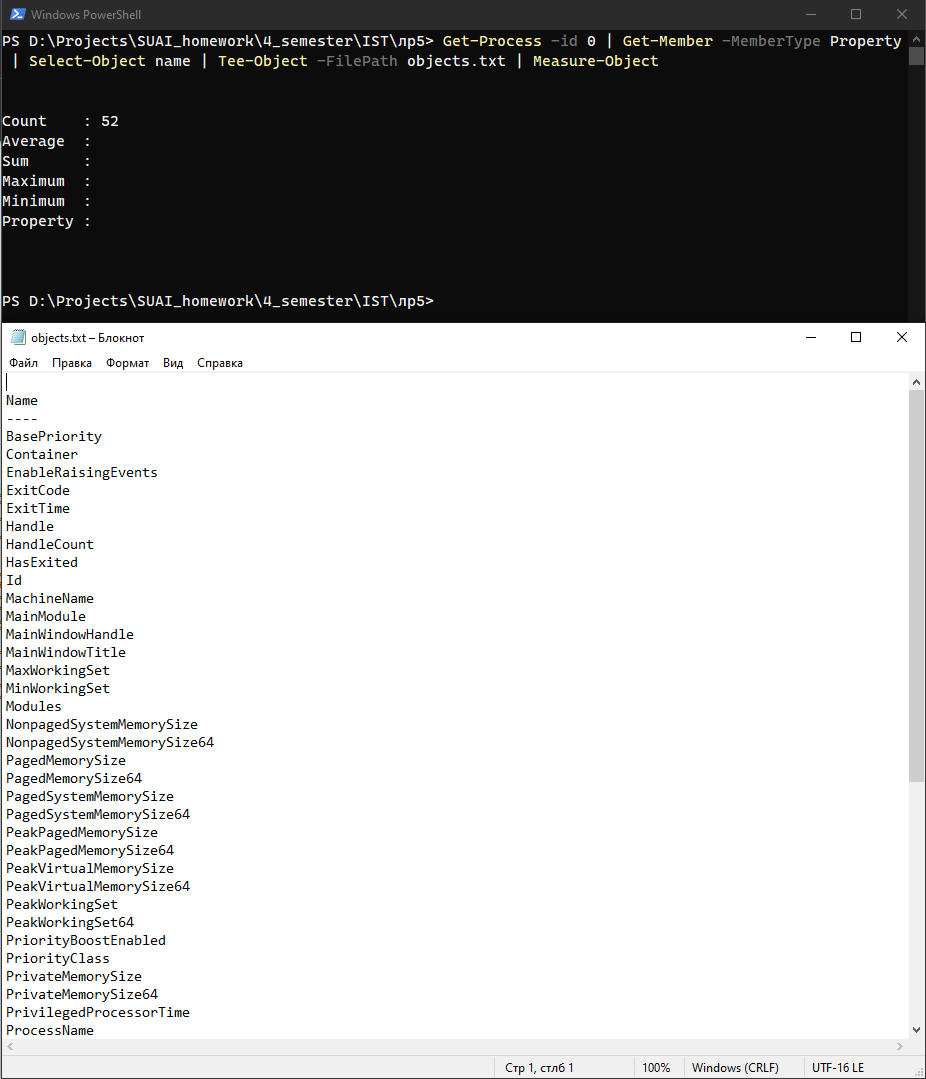
****

Рисунок 14 – Подсчет количества объектов и вывод результата в файл

## **2.6 Упражнение 2.10. – Чтение содержимого текстовых файлов**

В этом упражнении нам предлагается попрактиковаться в считывании файлов. Для примера файла используем файл из прошлого упражнения (см. рис. 14). Выполним упражнении и приложим скриншот.

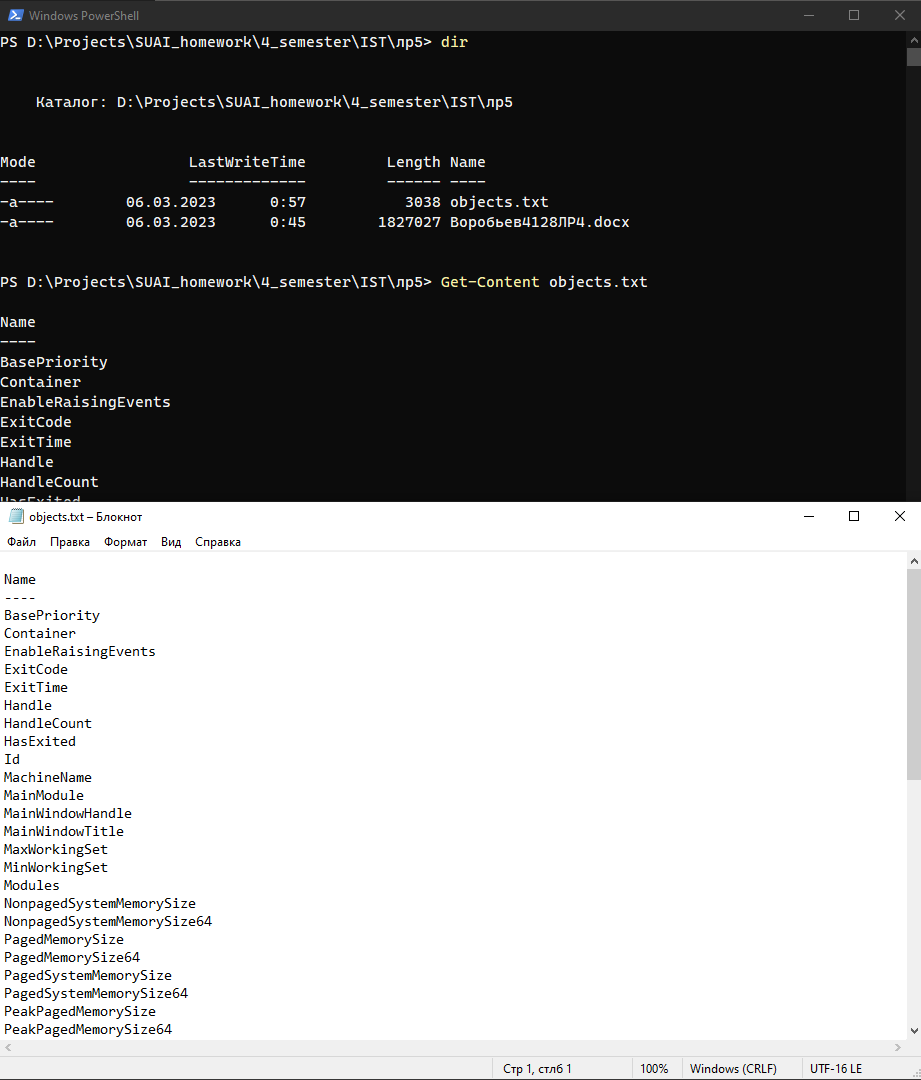
****

Рисунок 15 – Вывод содержимого файла

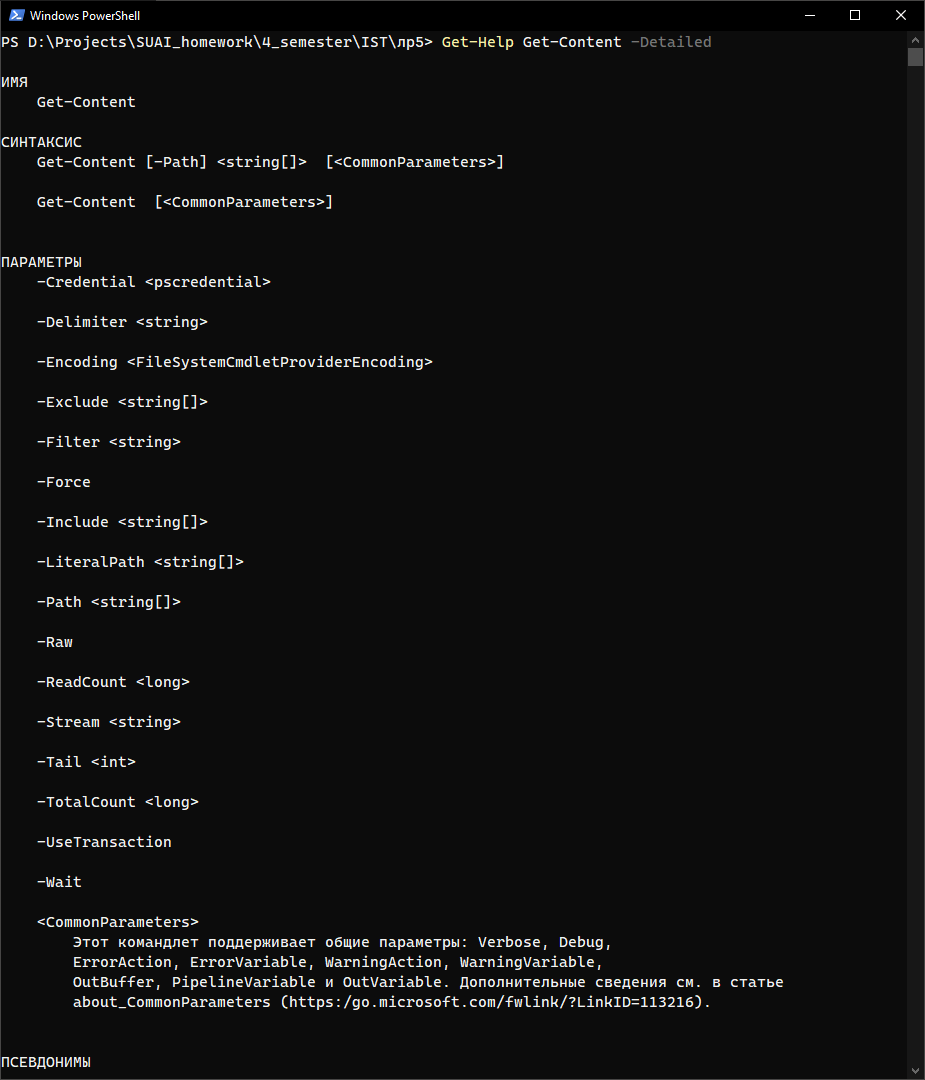


Рисунок 16 – Вывод справки по команде Get-Content

## **2.7 Упражнение 2.11. – Запись текста в файл**

В этом упражнении нам предлагается попрактиковаться в записи данных в файл. Выполним упражнение и приложим скриншоты.

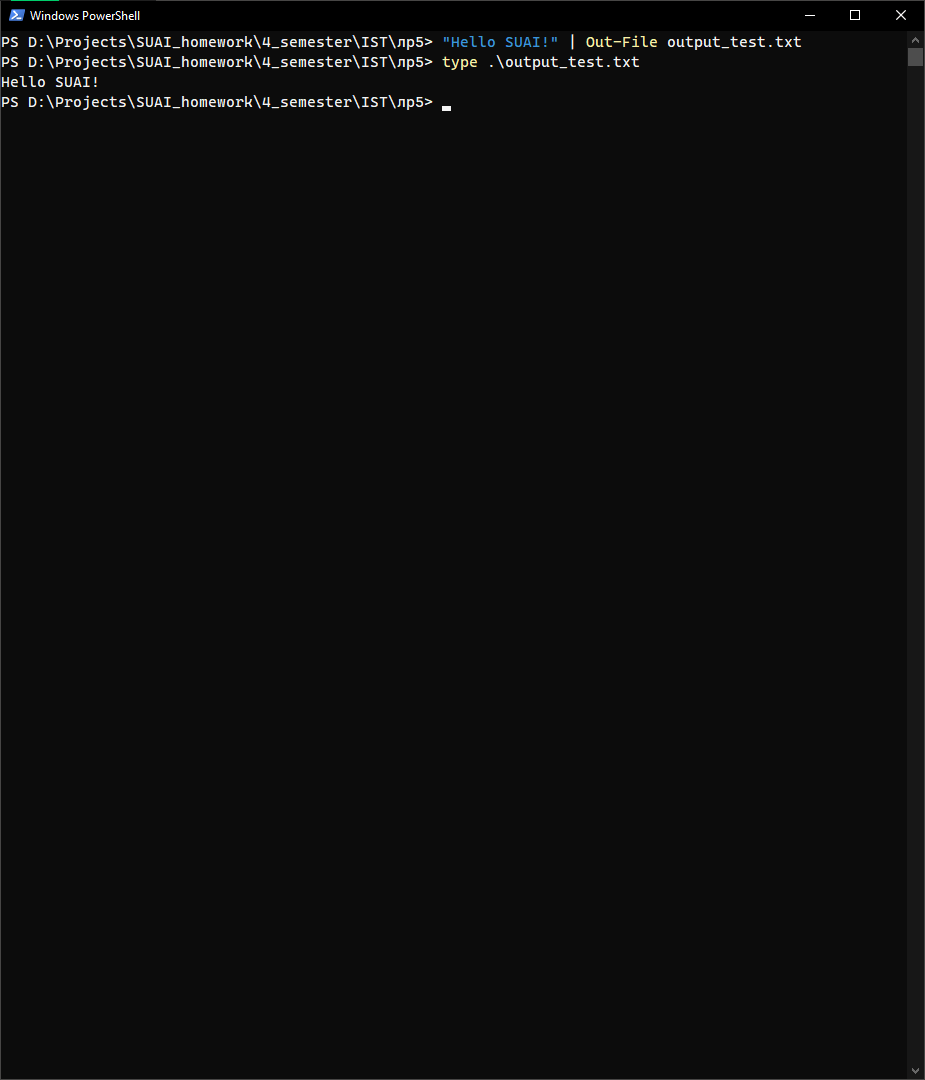


Рисунок 17 – Запись данных в файл

**3 Индивидуальное задание**

В индивидуальном задании от нас требуется изучить три командлета, а именно их возможности применения и привести примеры использования. В соответствии с нашим вариантом (см. рис. 1) мы должны выбрать командлеты: Format-Table, Stop-Process, Restart-Service.

## **3.1 Изучение Stop-Process**

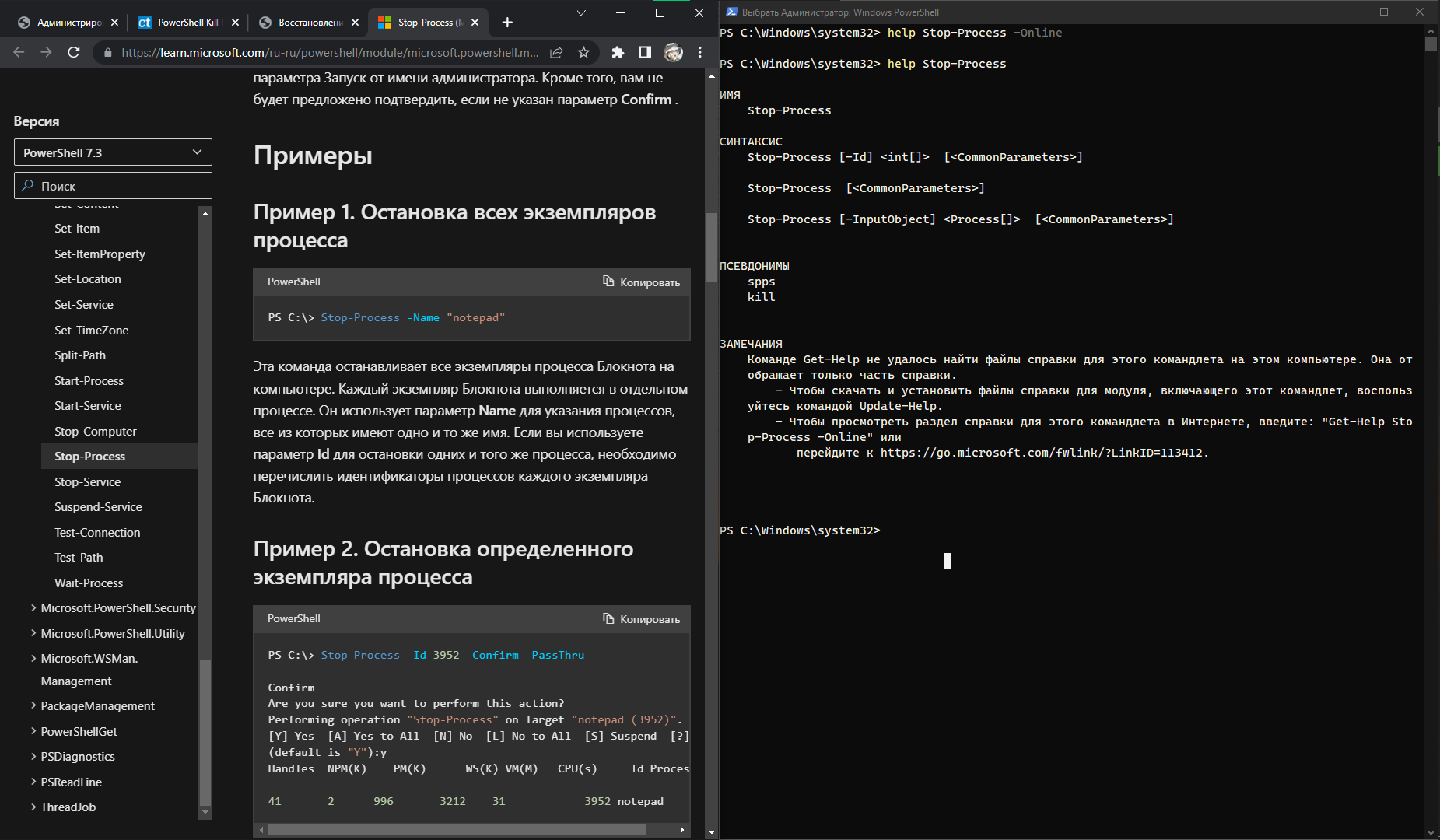


Рисунок 18 – Справка по командлету Stop-Process

Как понятно из справки Stop-Process используется для остановки одного или нескольких процессов. Наиболее важными для этого командлета считаю:

* -Id - Указывает идентификаторы процессов, которые необходимо остановить. При указании нескольких идентификаторов разделяйте их запятыми. По умолчанию удалению происходит по этому ключу.
* -Name - Указывает имена процессов, которые необходимо остановить.
* -Force - Останавливает указанные параметры без запроса подтверждения. По умолчанию запрашивает подтверждение перед остановкой любого процесса, Stop-Process который не принадлежит текущему пользователю.
* - Confirm - Запрос подтверждения перед выполнением командлета.

Протестируем командлет с приведёнными выше ключами.

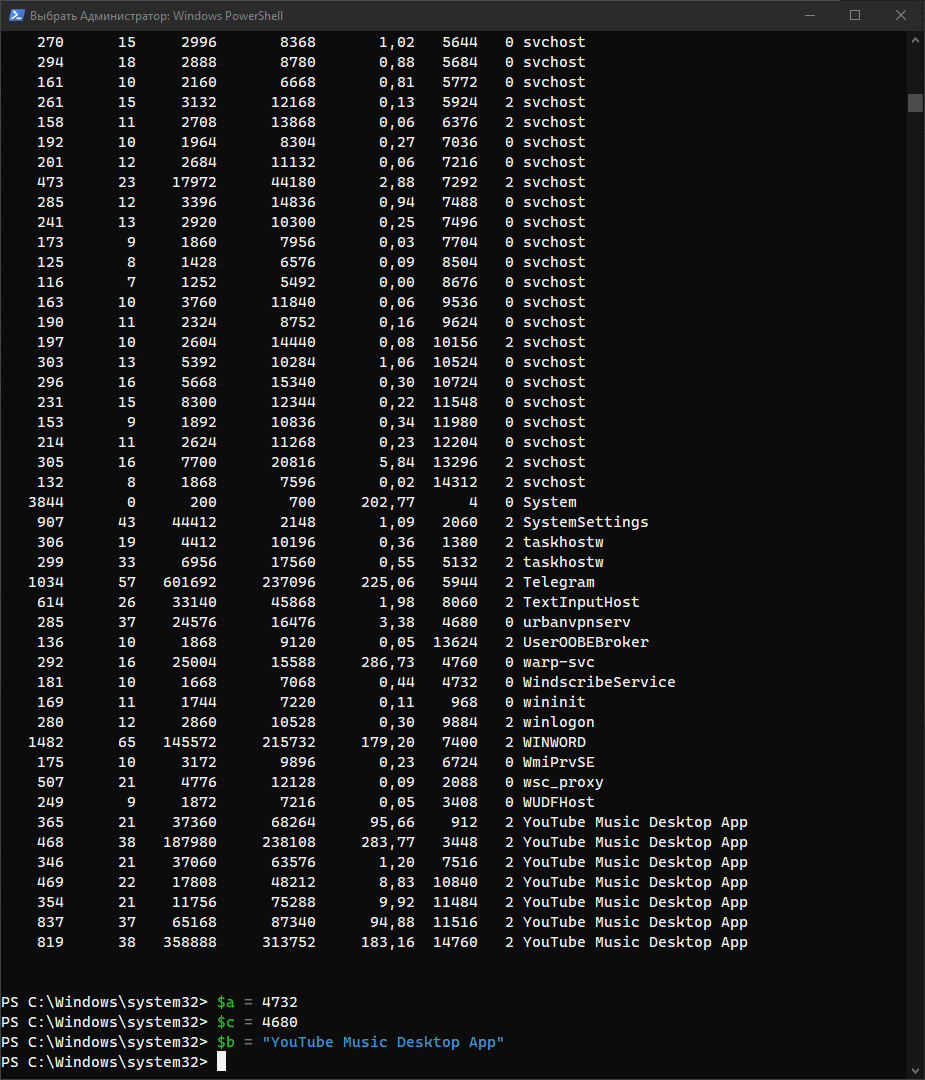


Рисунок 19 – Пример останавливаемых процессов

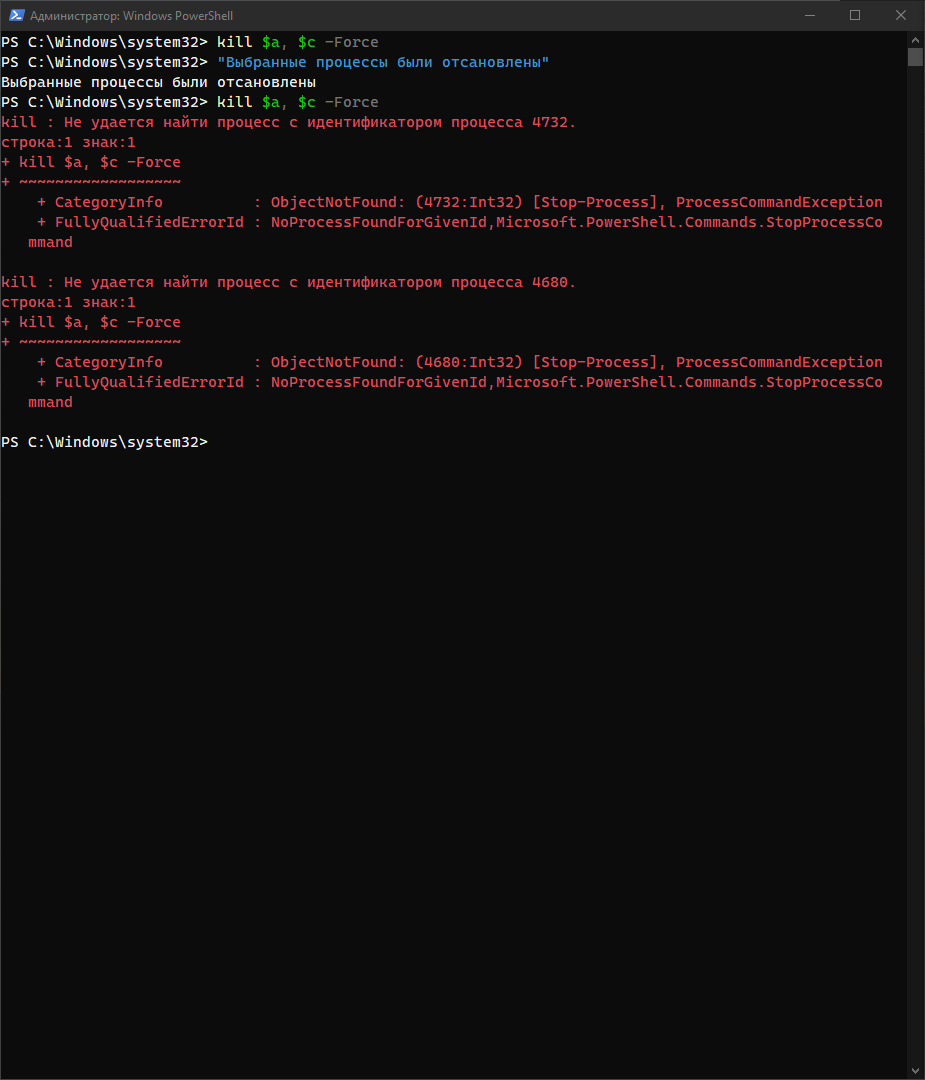


Рисунок 20 – Результат Stop-Process по ID с -Force

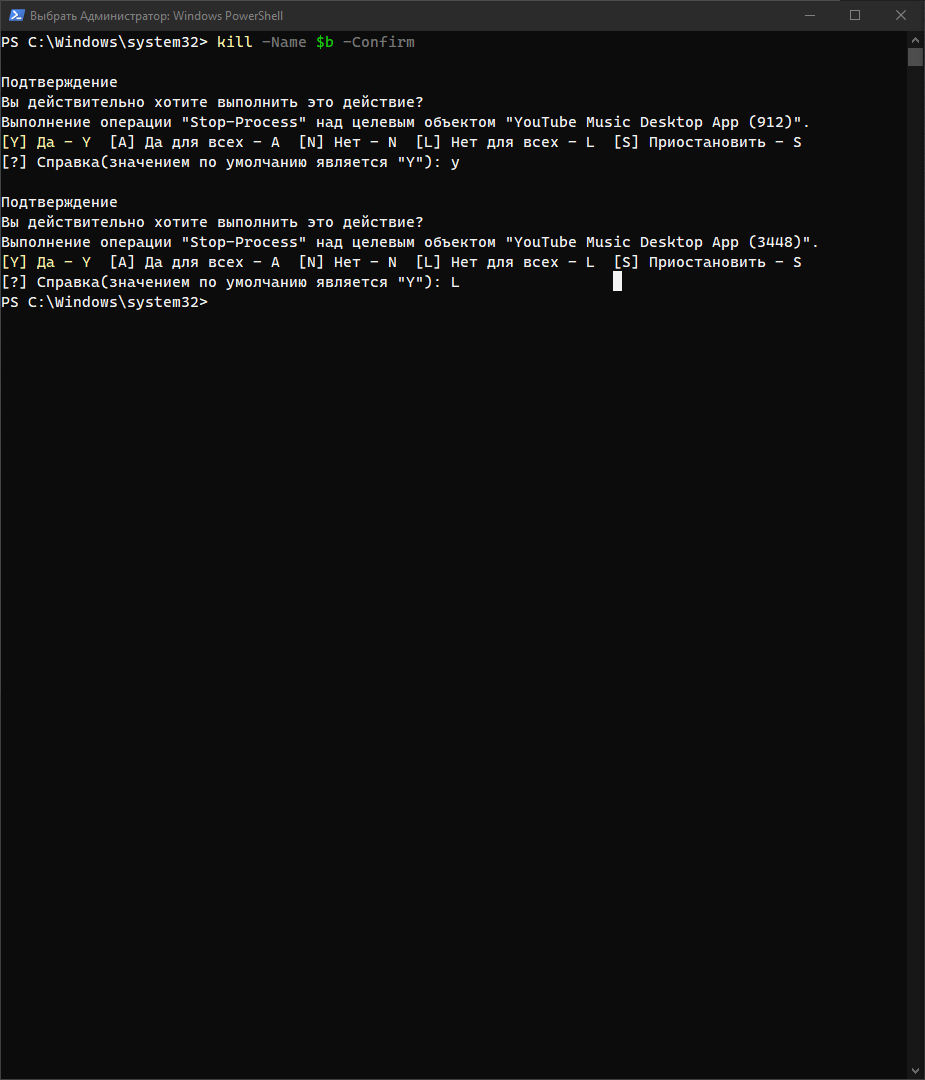


Рисунок 21 – Результат Stop-Process с -Force и -Name

**3.2 Изучение Restart-Service**

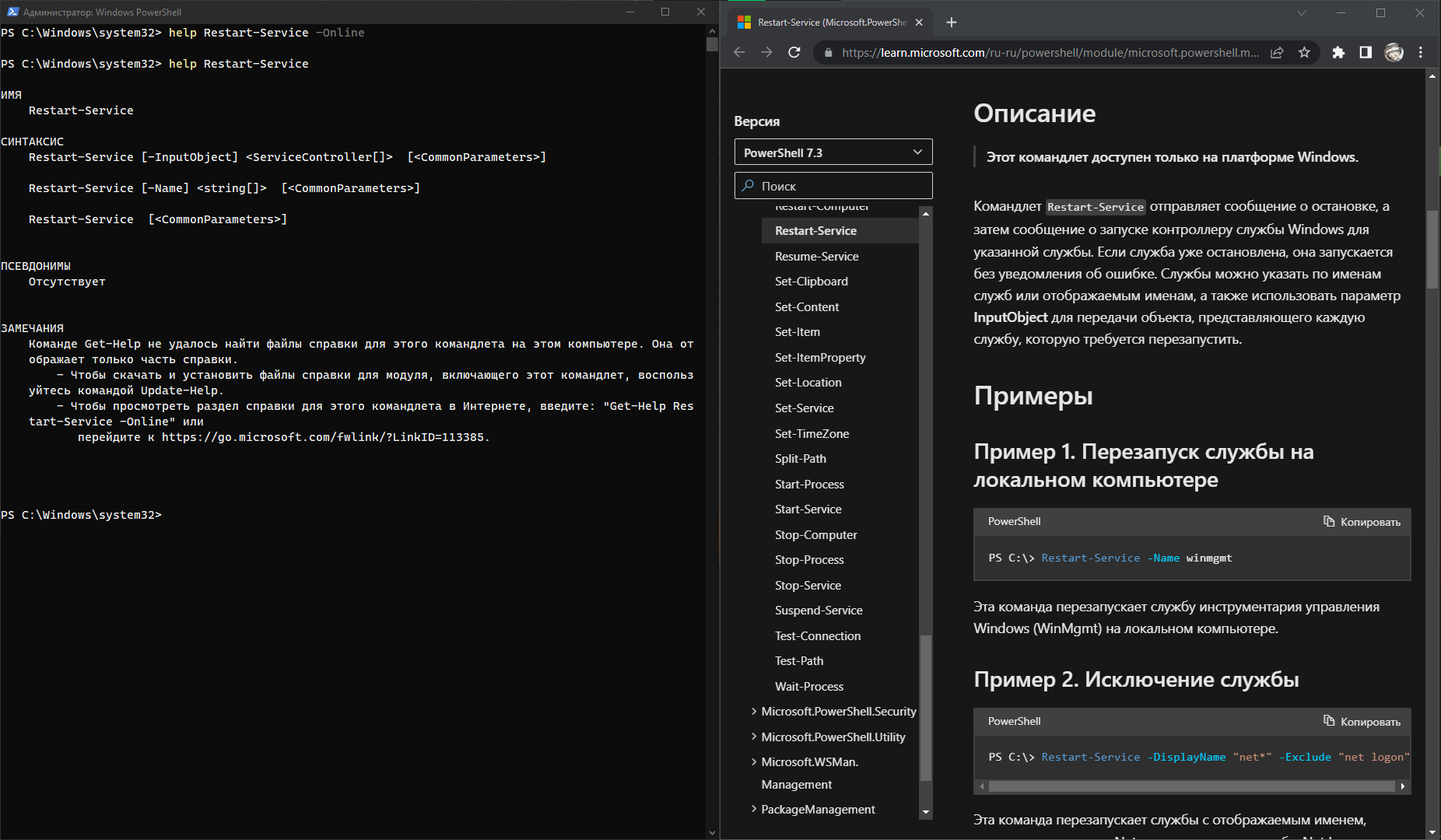


Рисунок 22 – Справка по Restart-Service

Как видно из справки командлет Restart-Service используется для перезапуска служб. Протестируем команду, используя следующие ключи:

* -DisplayName - Указывает отображаемые имена перезапускаемых служб. Можно использовать подстановочные знаки.
* -Force - Принудительное выполнение команды без запроса на подтверждение пользователем.
* -Exclude - Указывает службы, которые пропускает этот командлет. Указывает службы, которые пропускает этот командлет.
* -Name - Указывает имена перезапускаемых служб.

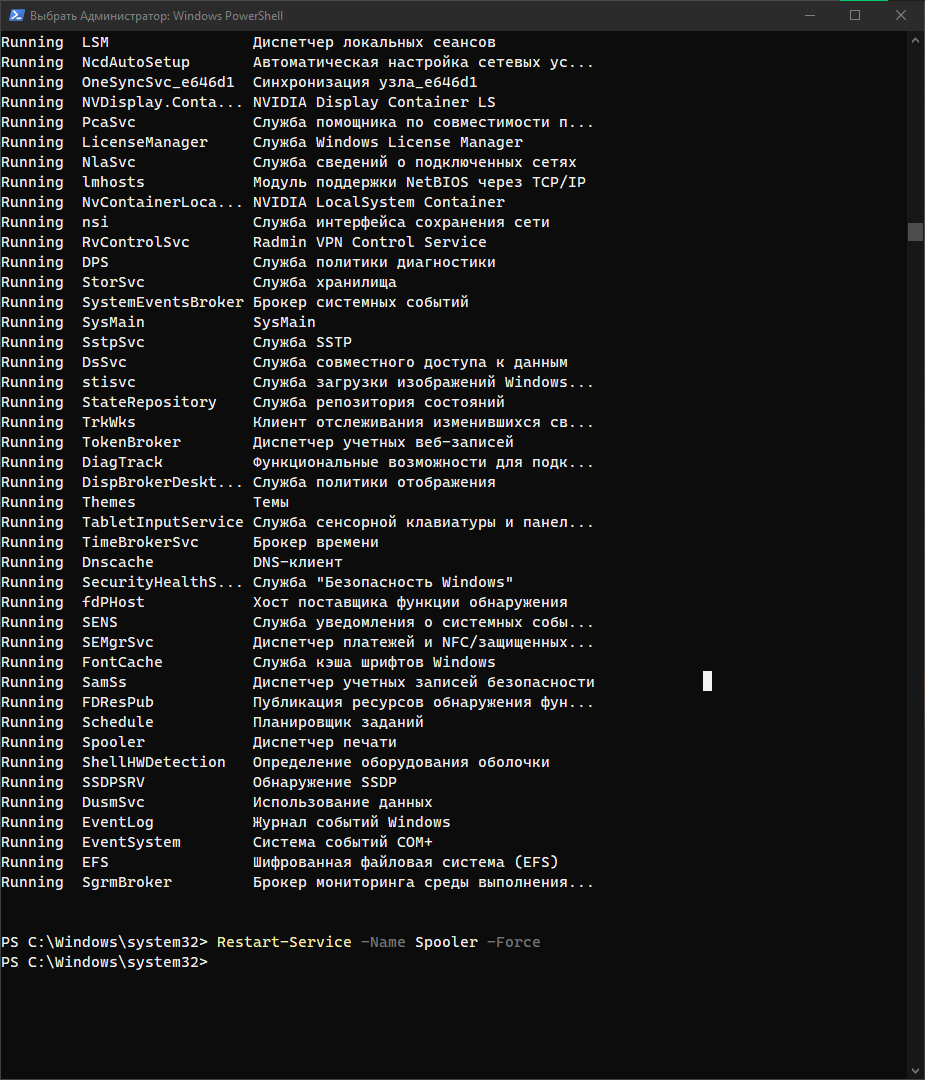


Рисунок 23 – Результат Restart-Service с -Name и -Force

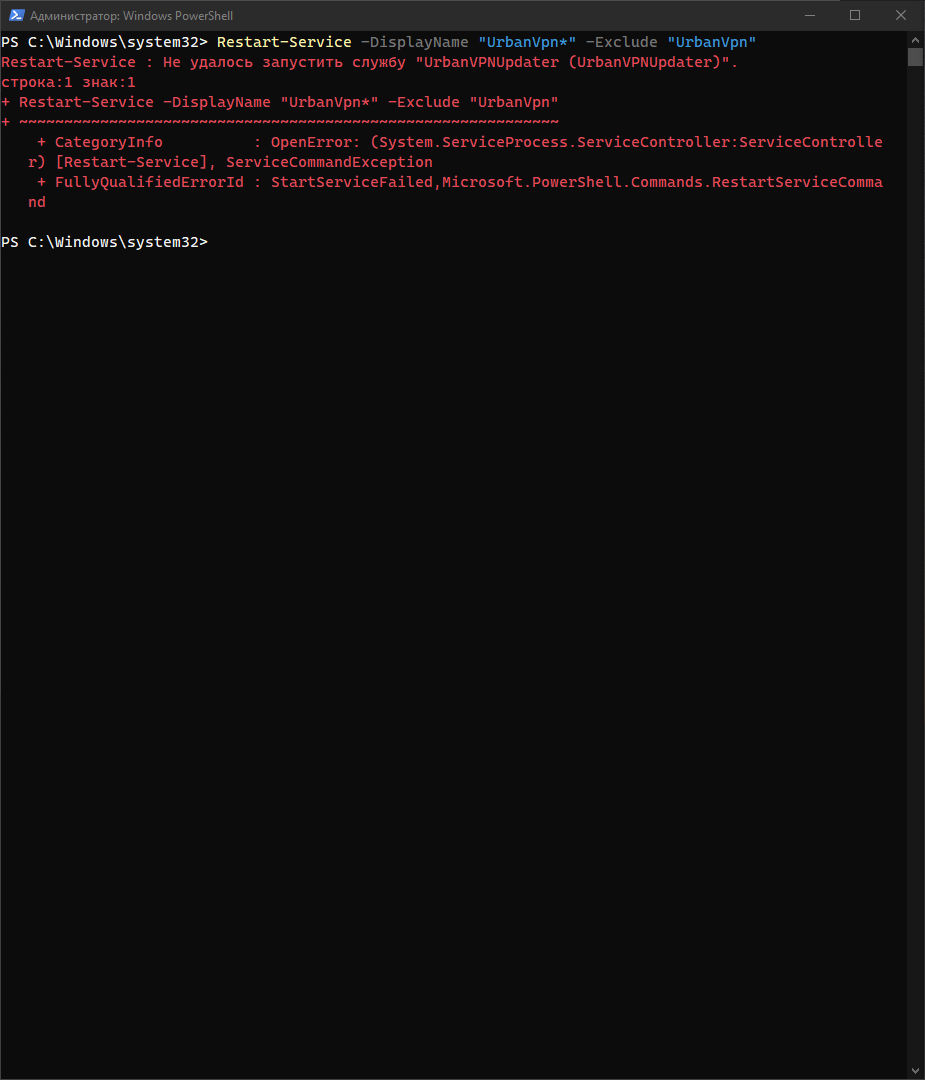


Рисунок 24 – Результат Restart-Service с -DisplayName и -Exclude

**3.3 Изучение Format-Table**

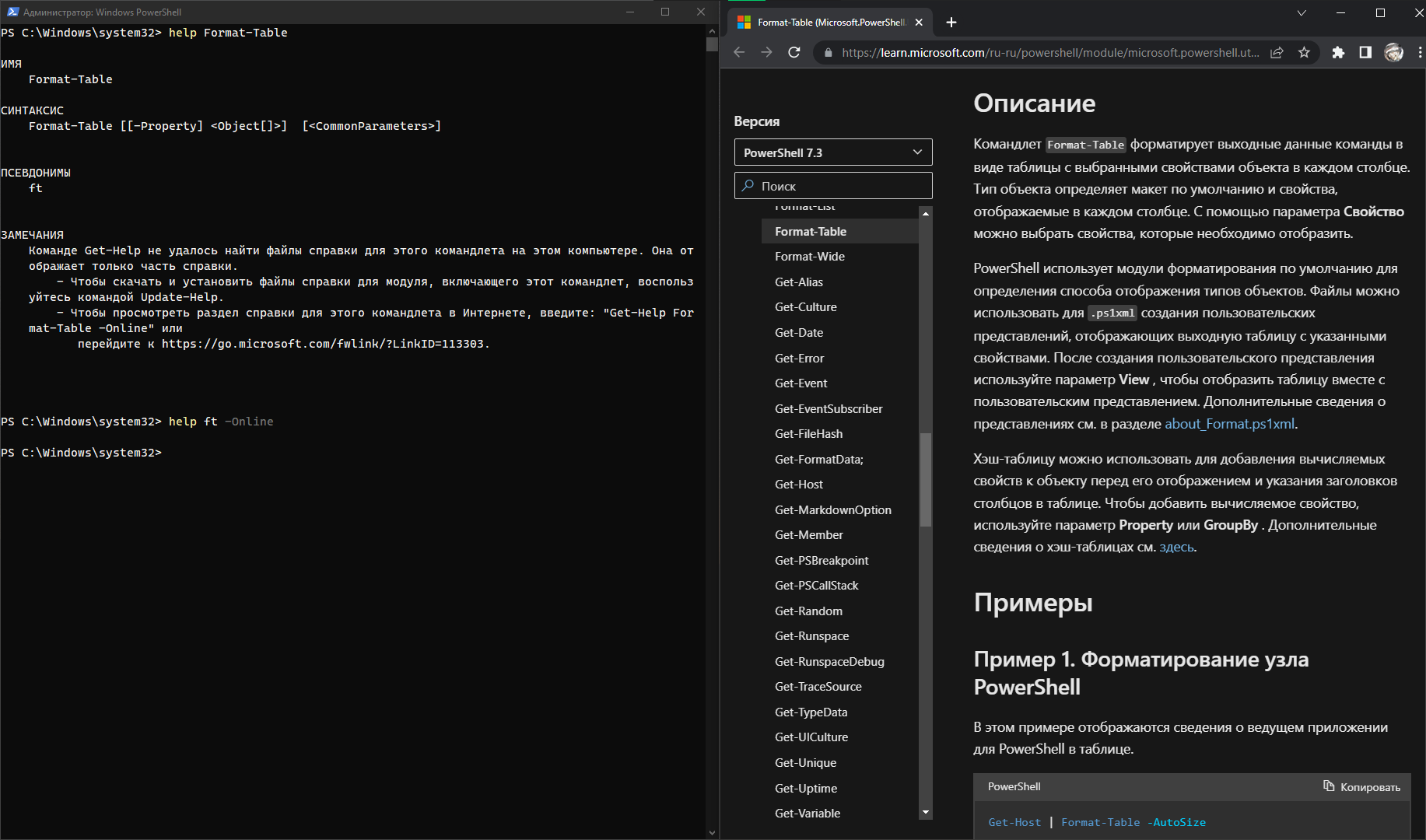


Рисунок 25 – Справка по команде Format-Table

Как видно из справки Format-Table форматирует выходные данные в виде таблицы. Команда очень простая и тем не менее, очень полезная для ясности отображения выходных данных. Для этой цели используются ключи:

* -GroupBy - Указывает отсортированные выходные данные в отдельных таблицах на основе значения свойства.
* -Property - Задает свойства объекта, которые будут включены в вывод, и порядок их вывода. Введите одно или несколько имен свойств, разделенных запятыми. Разрешено использовать подстановочные знаки.

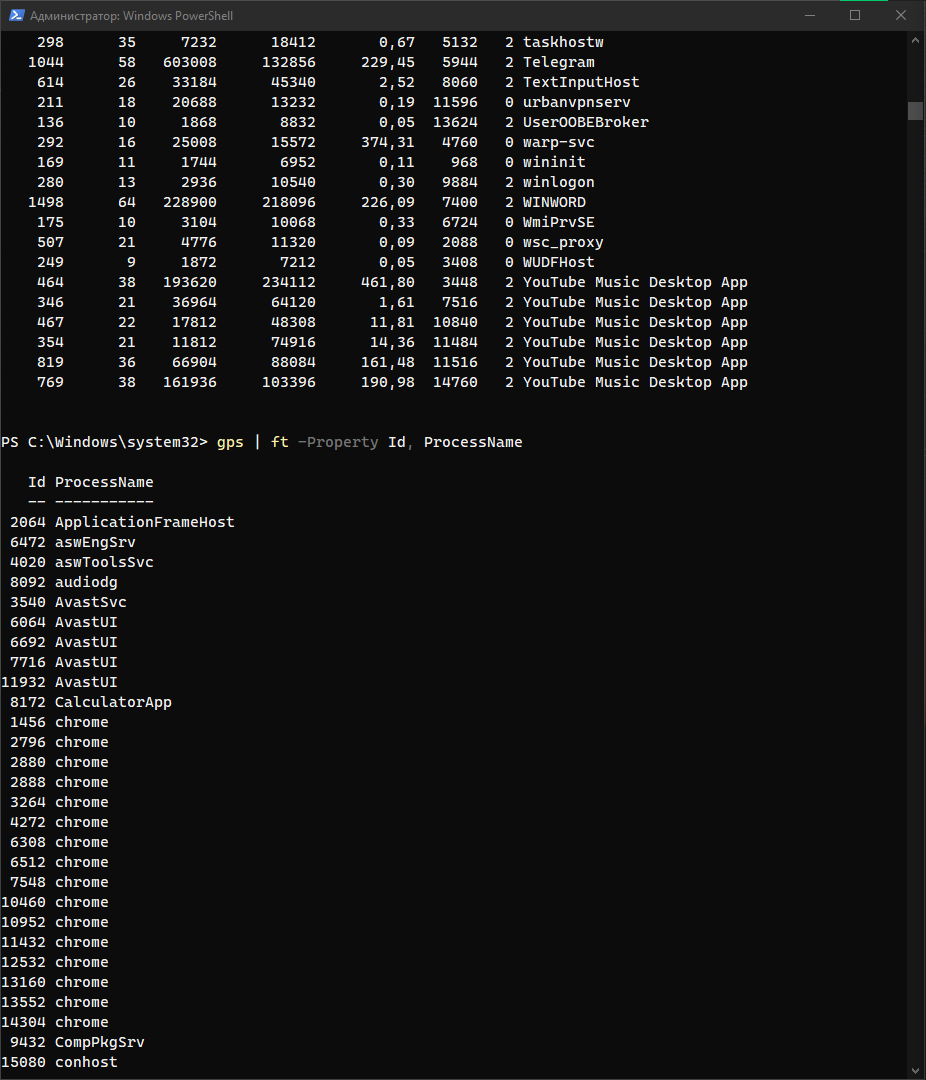


Рисунок 26 – Результат Format-Table с -Property

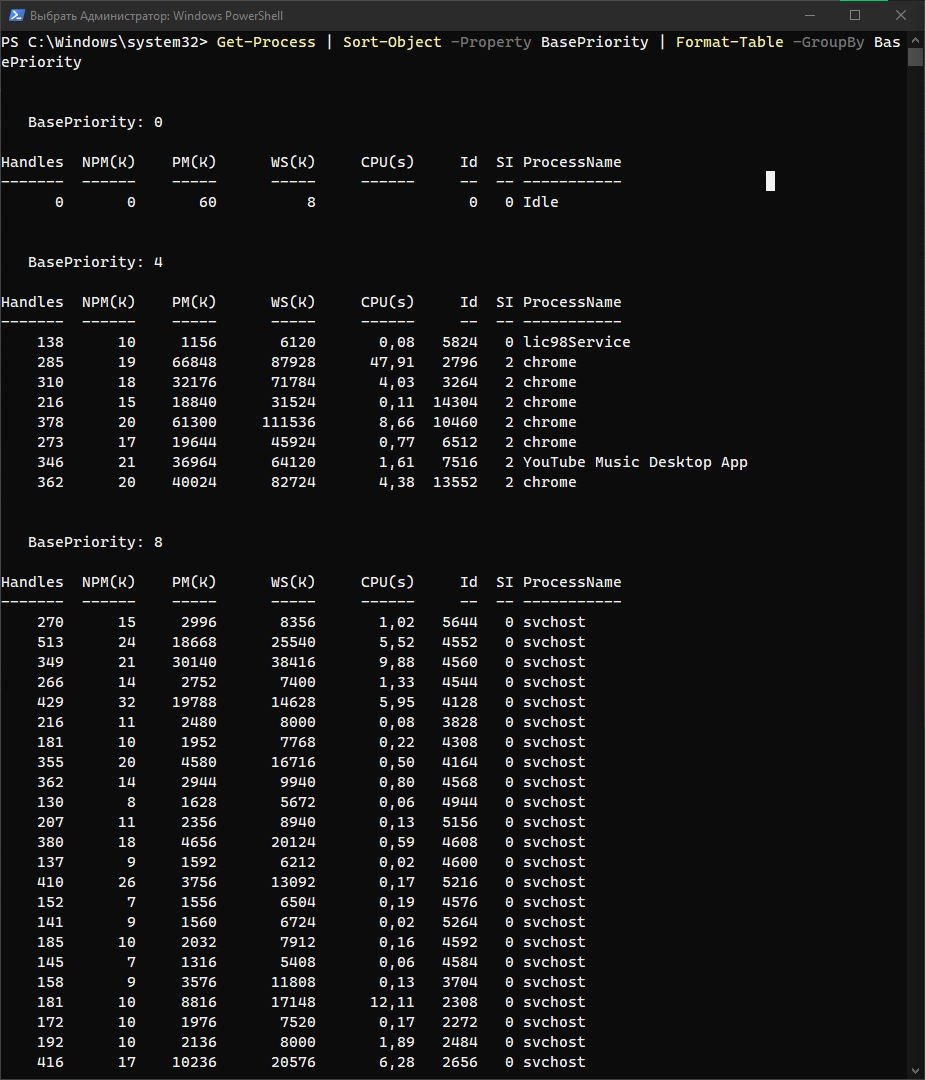


Рисунок 27 – Результат Format-Table с -GroupBy

**4 Сведения о системе**

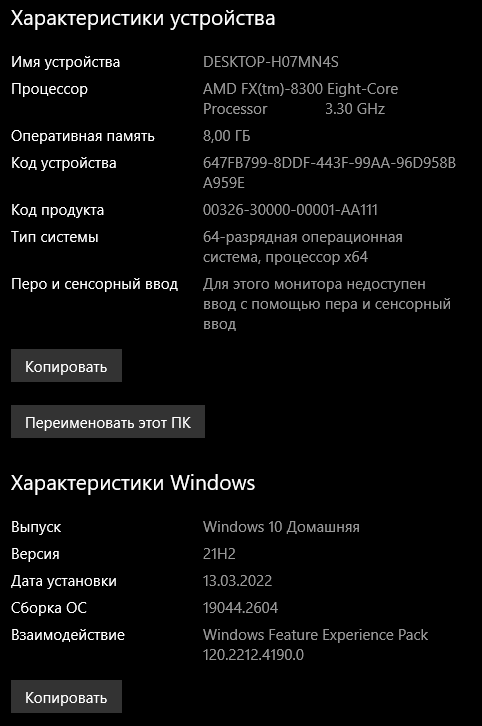


Рисунок 28 – Скриншот характеристик Windows

**5 Вывод**

В ходе выполнения лабораторной были изучены командлеты PowerShell, а также усовершенствованы навыки администрирования ОС Windows с помощью инструмента PowerShell.

Командлет — это специальный вид программы, предназначенной для автоматизации рутинных задач и запуска процессов в операционных системах Windows. Командлеты позволяют выполнять простые и сложные задачи администрирования.

В ходе выполнения упражнений были рассмотрены такие командлеты:

* Get-Process командлет позволяет пользователям просматривать и обрабатывать процессы, запущенные на их компьютере. Командлет предоставляет пользователям возможность просматривать информацию о запущенных процессах, включая имя процесса, идентификатор процесса, время запуска процесса, используемую память и другую информацию.
* Get-Service возвращает информацию о службах, запущенных на компьютере. Команда возвращает имя службы, ее состояние, а также другие детали, такие как имя пользователя, под которым запущена служба, и путь к файлу исполняемого модуля.
* Sort-Objects позволяет пользователям сортировать объекты по определенным параметрам. Это полезно для анализа данных и поиска определенных элементов. С помощью Sort-Objects можно сортировать по имени, типу, дате и другим параметрам. Это позволяет пользователям быстро находить и анализировать интересующие их данные.

В ходе выполнения индивидуального задания были рассмотрены командлеты Restart-Service, Stop-Process и Format-Table. Так как данные командлеты уже были рассмотрены в лабораторной работе, то ограничимся кратким описанием назначения команд:

* Restart-Service позволяет пользователям перезапускать службы Windows. Этот инструмент может быть использован для перезапуска служб, которые были остановлены или приостановлены.
* Stop-Process позволяет пользователям завершать процессы в Windows PowerShell. Это может быть полезно, если процесс не реагирует на запросы или приводит к проблемам производительности.
* Format-Table позволяет выводить информацию в табличном виде, тем самым повышая читабельность.

С помощью изученных команд можно выполнять администрирование, устранение ошибок и анализ производительности компьютеров.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. MicrosoftLearn: Документация PowerShell: сайт. – URL: https://learn.microsoft.com/ru-ru/powershell/ (дата обращения: 06.03.2023)
2. Администрирование и диагностика ОС Windows на персональном компьютере: учебное пособие/ А.В. Аграновский, К.Б. Гурнов, В.С. Павлов, Е.Л. Турнецкая. – СПб.: ГУАП, 2020. ‒ 148 с., ил. (дата обращения: 06.03.2023)
3. PC&Network: Команда Stop-Process: сайт. – URL: https://www.pcwdld.com/powershell-kill-process-command (дата обращения: 06.03.2023)
4. Comparitech: Powershell Kill: сайт. – URL: https://www.comparitech.com/net-admin/powershell-kill-process-command-tutorial/ (дата обращения: 06.03.2023)
5. FixMyPC: Команда Format-Table: сайт. – URL: https://fixmypc.ru/post/powershell-format-table-formatirovanie-vyvoda-v-tablitsu/ (дата обращения: 06.03.2023)
6. LinuxHint: Перезапуск службы: сайт. – URL: https://linuxhint.com/restart-service-using-powershell/ (дата обращения: 06.03.2023)