|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

**Отчет по практическим работам №5-9**

по дисциплине «Системное программное обеспечение»

|  |  |
| --- | --- |
| **Выполнил:**  Студент группы ИВБО-08-22 | Стецюк В.В. |
| **Проверил:** | ст. преподаватель Воронцов Ю.А. |

МОСКВА 2024 г.

1 Практическая работа 5

**Задание**

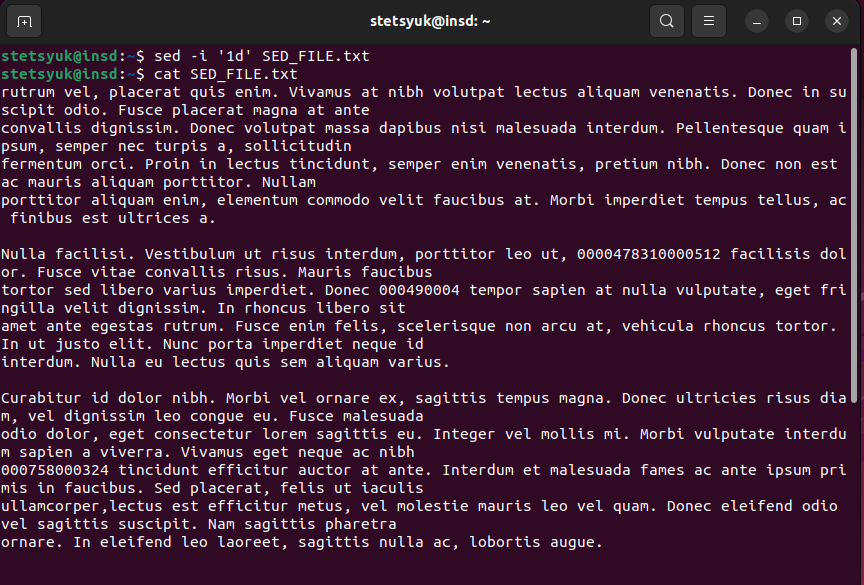
Используя утилиту sed выполнить следующие задачи:

* Удалить n-ую строку.
* Удалить все пустые строки.
* Удалить все строки до первой пустой строки, включительно.
* Вывести строки, содержащие определенное слово.
* В каждой строке, заменить любое первое встретившееся слово на слово "Linux".
* В каждой строке, заменить одно слово на другое по всему тексту.
* Удалить все пробелы в конце каждой строки.
* Заменить все последовательности ведущих нулей одним символом "0".
* Удалить все строки, содержащие определенное слово.
* Удалить все найденные вхождения выбранного слова, оставляя остальную часть строки без изменений.

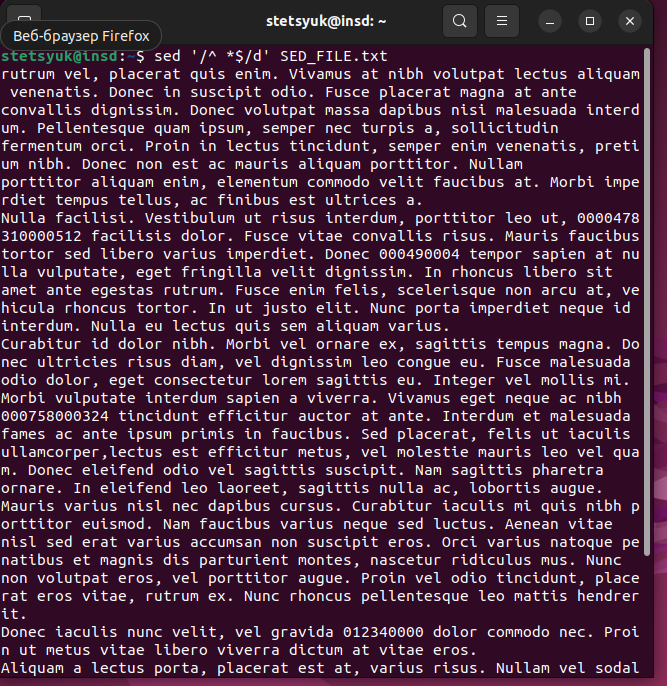
При помощи утилит sed реализовать поиск информации, хранящейся в файле MOCK\_DATA.csv. Результаты должны выводиться в файл, в вывод необходимо записать найденные строки с их номерами.

|  |  |
| --- | --- |
| **№** | **Искомое значение** |
| 1 | IP со значением 31 в третьем октете |
| 2 | URL в домене uk |
| 3 | Файлы с расширением mov |
| 4 | Email адрес в домене amazon |
| 5 | IP со значением 141 во втором октете |
| 6 | URL в домене edu |
| 7 | Файлы с расширением txt |
| 8 | Email адрес в домене yahoo |
| 9 | IP со значением 95 в четвертом октете |
| 10 | URL на файл с 3-им уровнем вложенности (пример: https://wiki.com/assets/images/image.jpg) |
| 11 | Файлы с расширением mpeg |
| 12 | Email адрес, содержащий число вида 9X |
| 13 | IP со значением 46 в первом октете |
| 14 | URL на файл с 4-ым уровнем вложенности (пример: https://wiki.com/assets/images/jpg/image.jpg) |
| 15 | Файлы с расширением gif |
| 16 | Email адрес, содержащий число 0 |

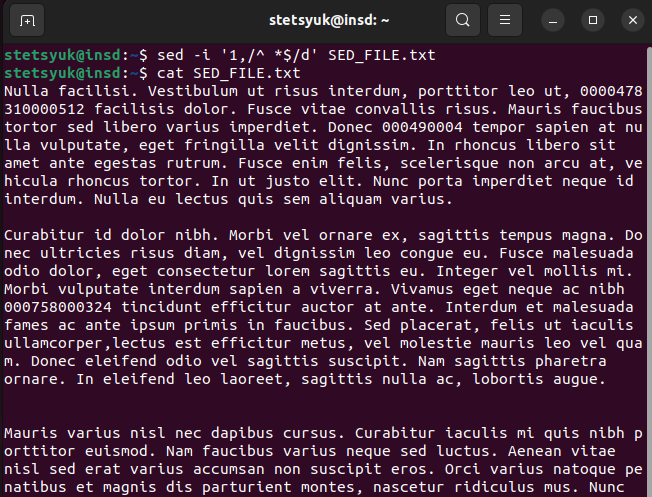
При помощи утилиты sed и команды d с указанием нужной строки можно удалить указанную строку.



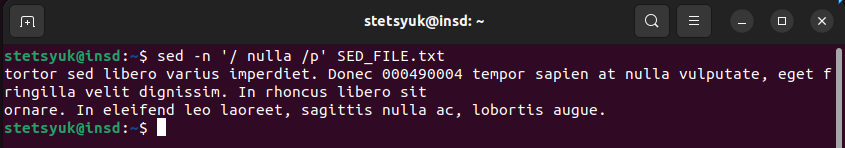
Для удаления пустых строк в файле воспользуемся командой sed, где ^ - определитель начала строки, $ - определитель конца строки, « \*»-команда ля поиска пробелов, d – команда для удаления строк. Шаблон обособим слешами.



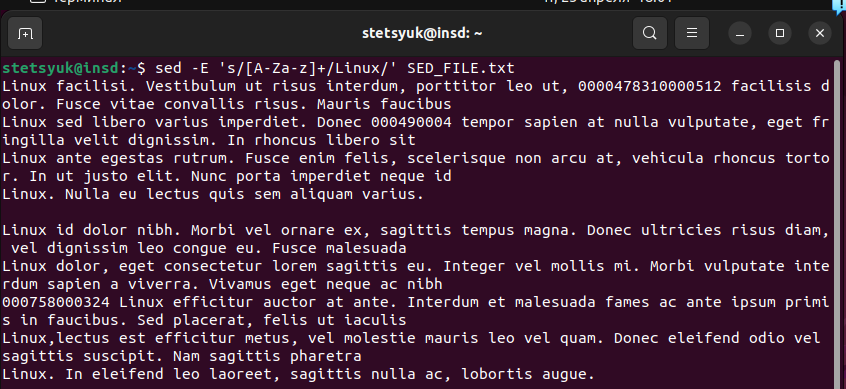
Чтобы удалить все строки до первой пустой, необходимо использовать специальную адресацию вида: 1,.



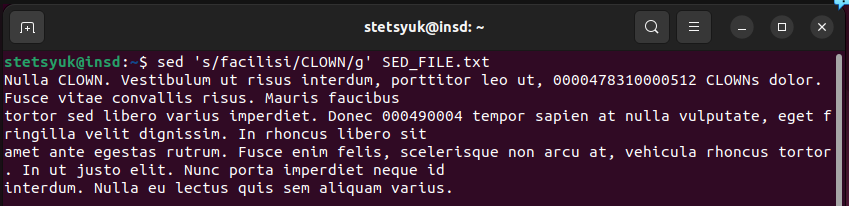
Для выполнения данной команды, потребуется ключ -n, который будет отменять вывод информации из буфера, а также команду p, которая выводит информацию в консоль.



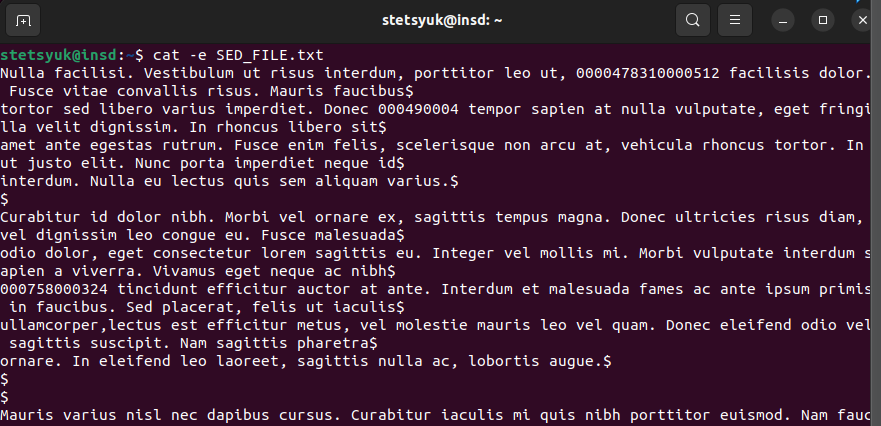
Для выполнения задания, необходимо использовать ключ -E, чтобы можно было использовать дополнения к регулярным выражениям.



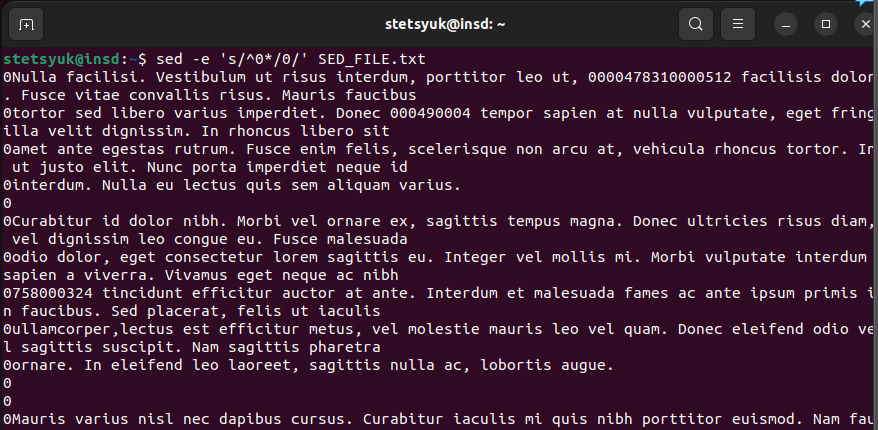
Чтобы заменить одно слово на другое, необходим параметр s и q. S – замена, g – все вхождения.



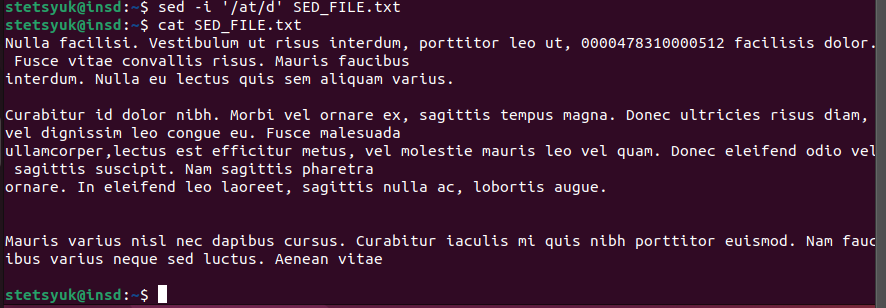
Для удаления пробелов в конце строки, воспользуемся заменой символов пробела на ничего.(**sed -i ‘s/ \*$//’ SED\_FILE.txt**)



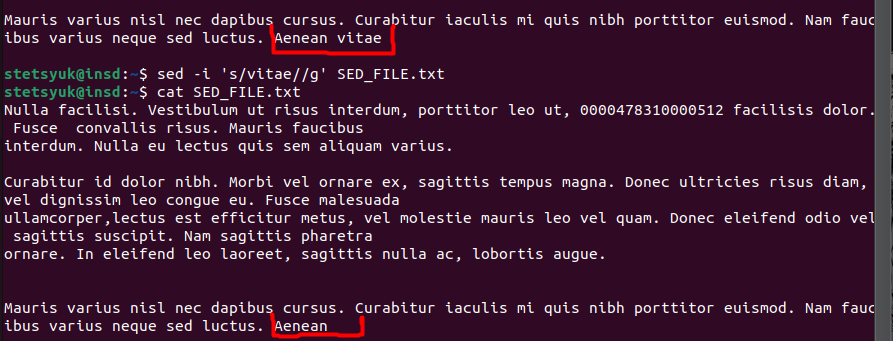
Замена ведущих нулей на один ноль



Чтобы удалить строки, содержащие определенное слово, необходимо использовать команду d.

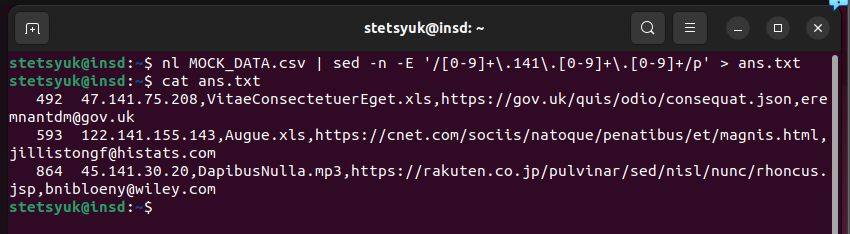


Это задание можно выполнить заменой нужного слова на пустую последовательность



Далее добавим файл MOCK\_DATA.csv на виртуальную машину и будем работать с ним.

Выборка по IP со значением 141 во втором октете находится так же.



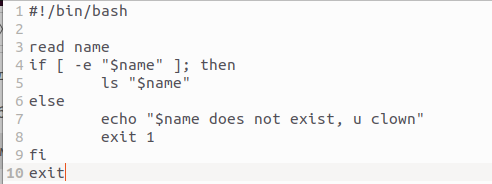
2 Практическая работа 6

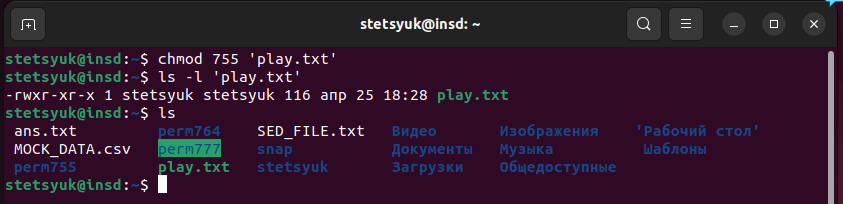
**Задание**

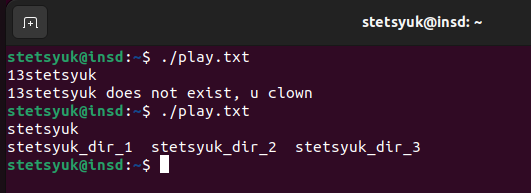
1. Напишите сценарий, который выводит содержимое любого каталога или сообщение о том, что его не существует.
2. Напишите сценарий, который с помощью цикла прочитает файл и выведет его содержимое.
3. Напишите сценарий, который с помощью цикла выведет список файлов и директорий из текущего каталога, укажет, что есть файл, а что директория.
4. Напишите сценарий, который подсчитает объем диска, занимаемого директорией. В качестве директории можно выбрать любую директорию в системе.
5. Напишите сценарий, который выведет список всех исполняемых файлов в директории, для которых у текущего пользователя есть права на исполнение.
6. При помощи циклов и функций реализовать поиск информации, хранящейся в файле MOCK\_DATA.csv. Результаты должны выводиться в файл, в вывод необходимо записать номер строки, а также найденное значение.

|  |  |
| --- | --- |
| **№** | **Искомое значение** |
| 1 | IP со значением 31 в третьем октете |
| 2 | URL в домене uk |
| 3 | Файлы с расширением mov |
| 4 | Email адрес в домене amazon |
| 5 | IP со значением 141 во втором октете |
| 6 | URL в домене edu |
| 7 | Файлы с расширением txt |
| 8 | Email адрес в домене yahoo |
| 9 | IP со значением 95 в четвертом октете |

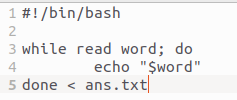
Для вывода содержимого каталога можно создать небольшой сценарий, который будет считывать название директории и если она существует, что проверяется при помощи условия, выводит её содержимое.

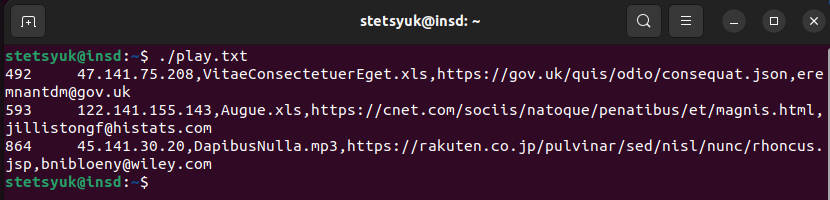




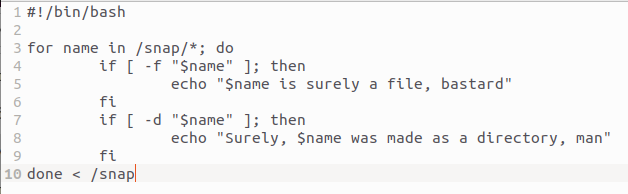


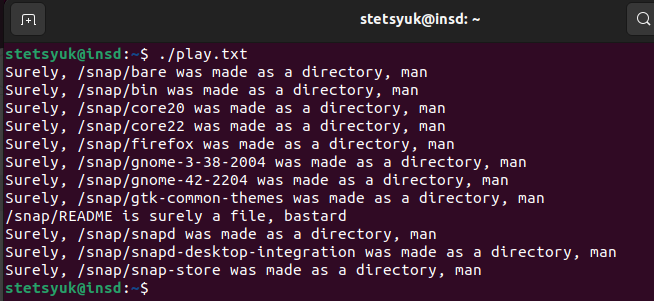
Далее напишем сценарий, который при помощи цикла выводит содержимое файла.



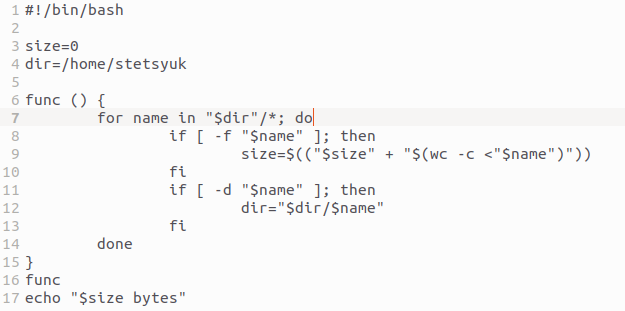


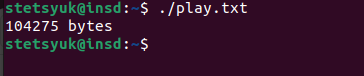
Для сценария, который с помощью цикла выведет список файлов и директорий из текущего каталога, укажет, что есть файл, а что директория, нужно воспользоваться условием.



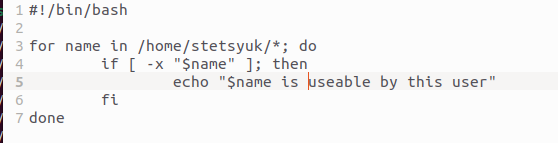


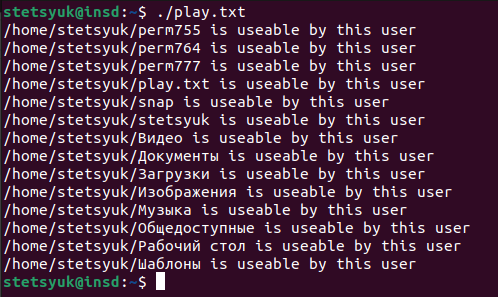
Для сценария, который подсчитает объем диска, занимаемого директорией, потребуется создать функцию, рекурсивно вычисляющую объём.



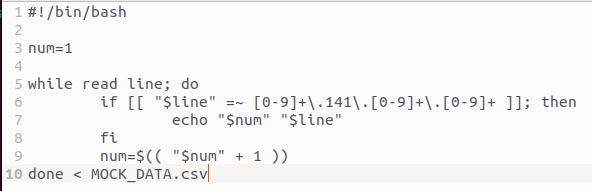


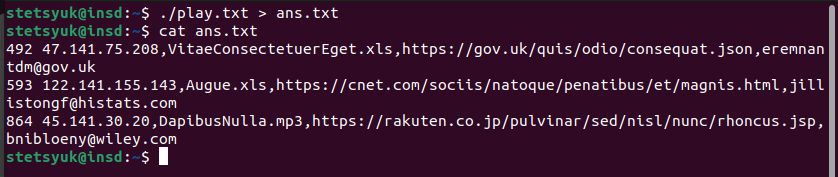
Для сценария, который выведет список всех исполняемых файлов в директории, для которых у текущего пользователя есть права на исполнение нужно воспользоваться –х в условии.





Далее при помощи циклов и условий выведем строки данных из файла MOCK\_DATA.csv аналогично предыдущей работе. Так же воспользуемся регулярными выражениями создавая из них необходимый шаблон.



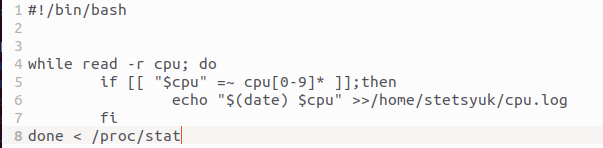
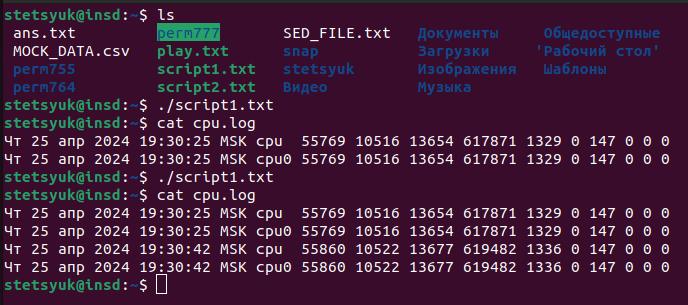


3 Практическая работа 7

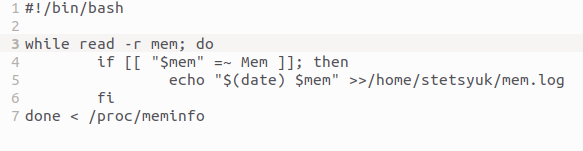
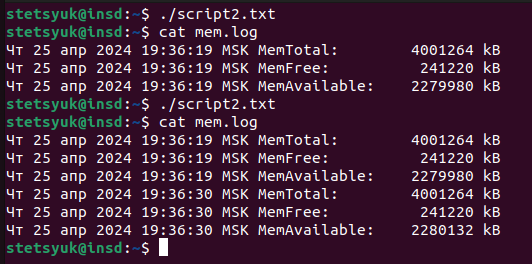
**Задание**

1. Создать скрипт логирования загруженности процессора. Каждая строка лога должна содержать время логирования, информацию об общей загруженности, а также информацию по загруженности каждого ядра процессора (необходимо выделить виртуальной машине хотя бы 2 ядра через настройки VirtualBox). Информацию о загруженности процессора можно найти в файле /proc/stat.
2. Создать скрипт логирования загруженности оперативной памяти. Каждая строка лога должна содержать время логирования, общий объем доступной памяти, объем свободной памяти, объем занятой памяти. Информацию о загруженности процессора можно найти в файле /proc/meminfo.
3. При помощи **cron** организовать запуск написанных скриптов по расписанию. Логгер загруженности процессора должен срабатывать раз в минуту в период с 9 до 18, и раз в 3 минуты в период 18 до 9. Логгер загруженности памяти должен срабатывать каждые 2 минуты период с понедельника по пятницу.

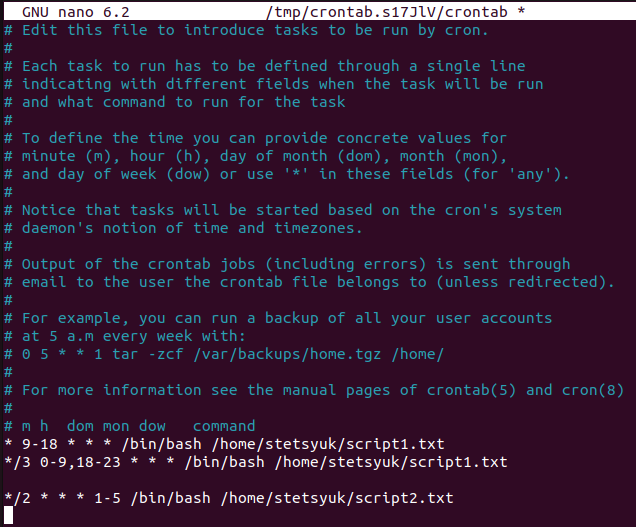
Создать скрипт загруженности процессора можно по аналогии с тем, как это делалось в предыдущей практике, информацию о загруженности процессора можно взять из специального файла /proc/stat.

Аналогично создаётся скрипт для логирования информации о загруженности оперативной памяти.

Для организации запуска созданных скриптов нужно отредактировать файл crontab при помощи команды crontab –e и выбора текстового редактора, так как команда запускается первый раз. В нём нужно указать необходимое время запуска скриптов.

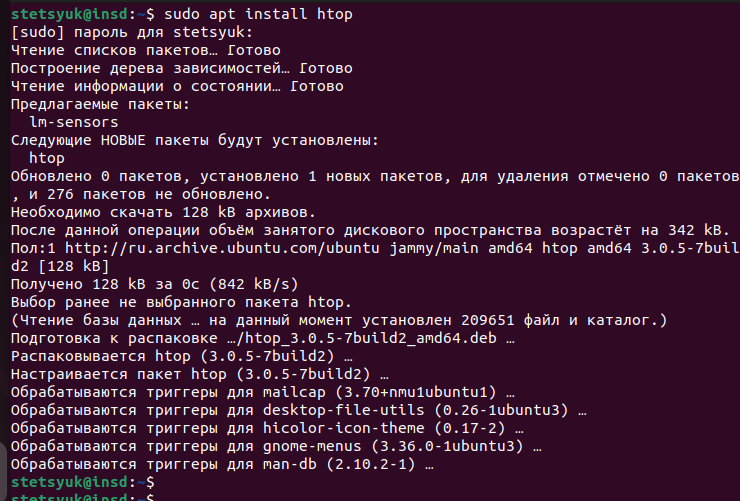


4 Практическая работа 8

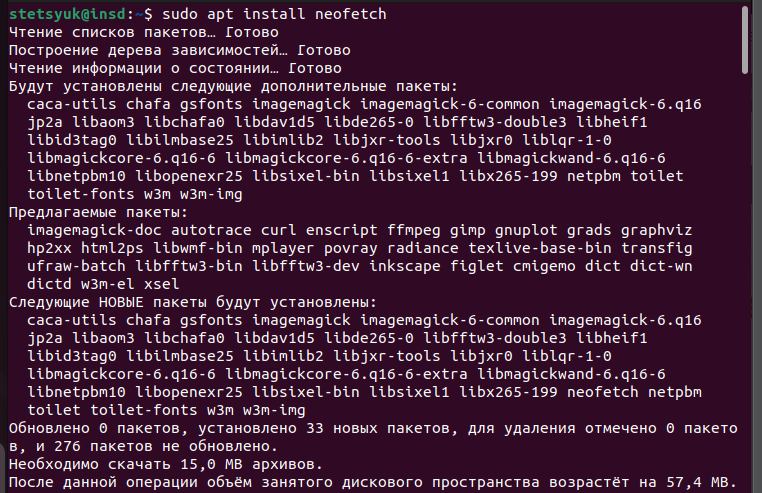
**Задание**

1. Установите пакеты **htop** и **neofetch** (запустите их после установки для проверки её корректности);
2. Удалите пакет **neofetch** без удаления конфигурационных файлов;
3. Найдите пакет **sqlite** третьей версии в дереве репозиториев, после чего выведите информацию о пакете;
4. Установите Docker, следуя официальной документации <https://docs.docker.com/engine/install/ubuntu/>.

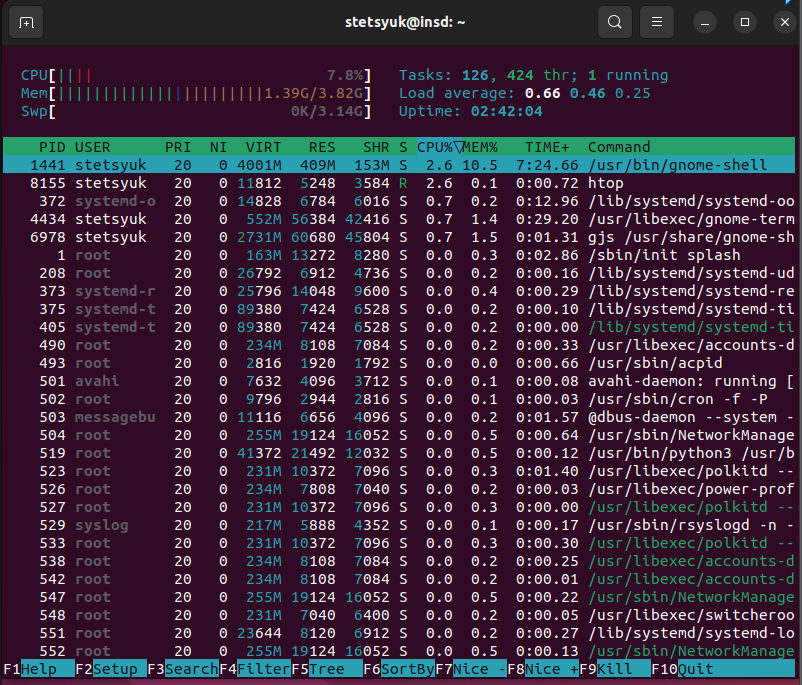
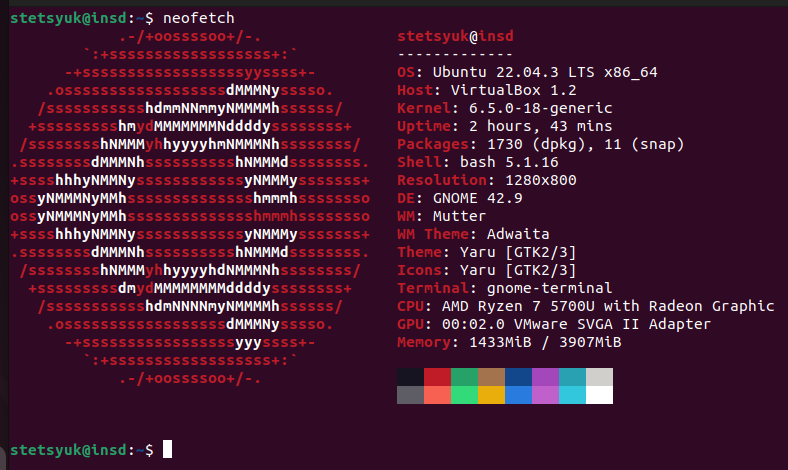
Для того, чтобы установить пакеты htop и neofetch нужно воспользоваться комндой apt install с правами корневого пользователя. Пакет успешно установлен.

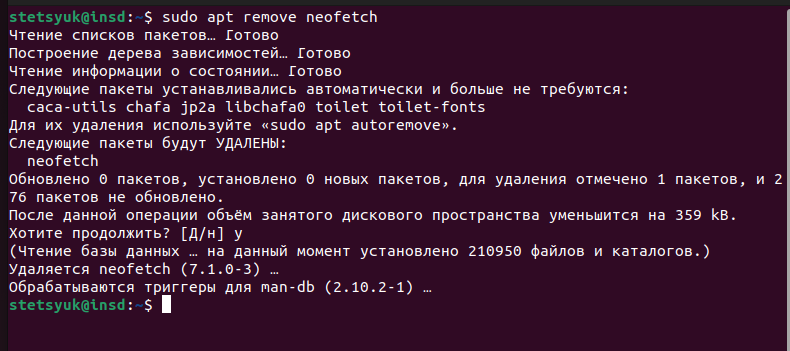


Аналогично устанавливается и пакет neofetch. Этот пакет потребовал небольшого разрешения на занятие небольшого пространства памяти.

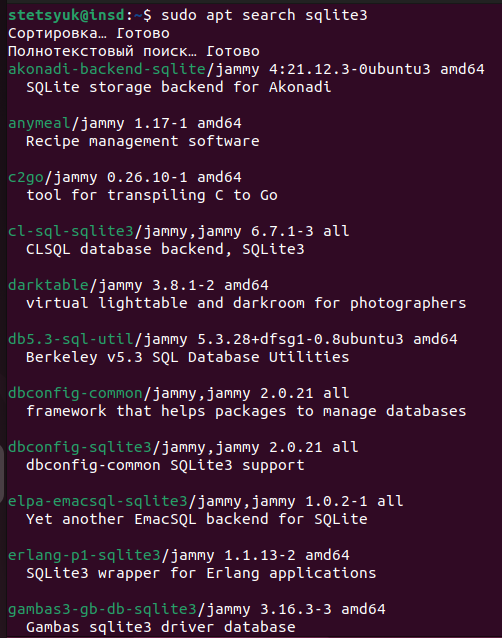


Оба пакета работают корректно.

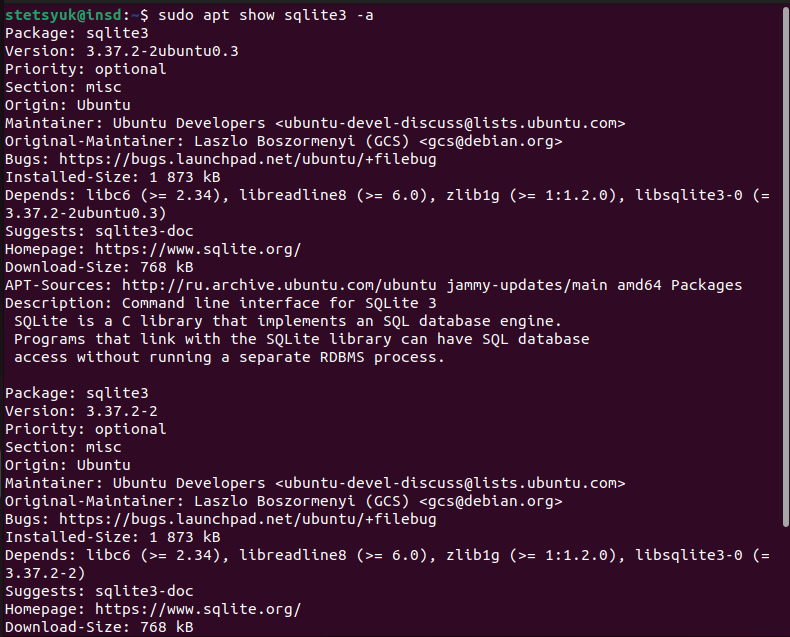
  Далее нужно удалить пакет neofetch без удаления конфигурационных файлов. Команда так же запрашивает подтверждения.



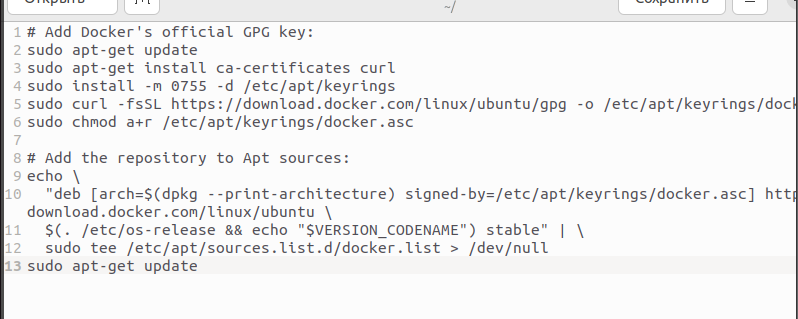
Чтобы найти пакет sqlite третьей версии воспользуемся командой apt search. Среди выведенных файлов есть и искомый пакет.

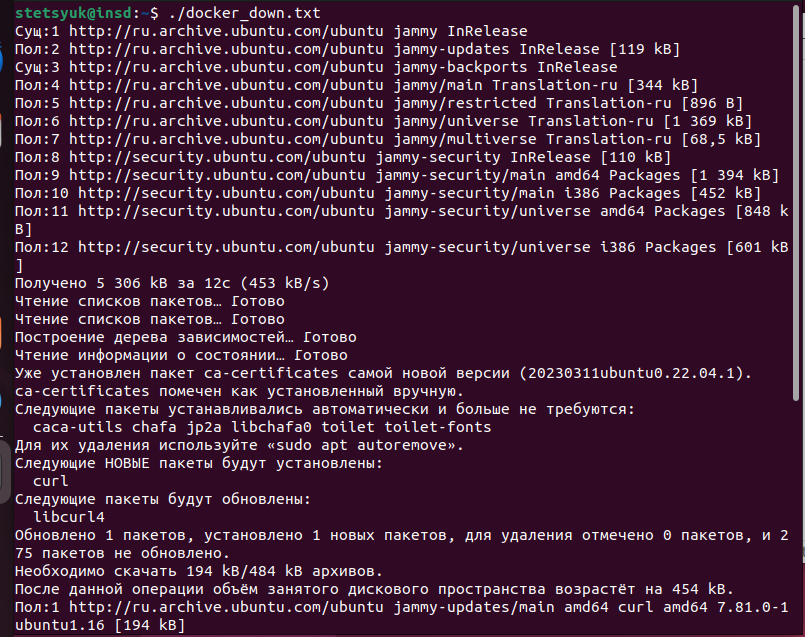


Информацию о пакете можно вывести при помощи apt show.

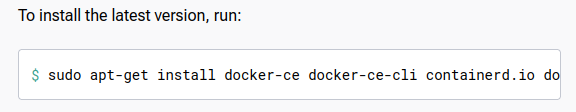


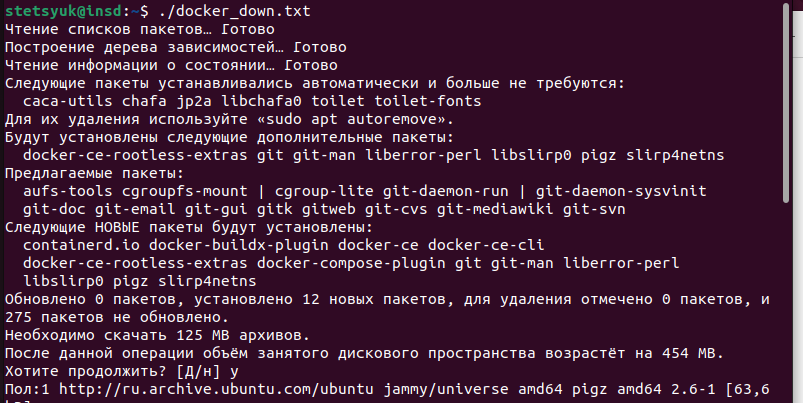
Для установки Docker будем следовать инструкциям. Изначально нужно установить репозиторий Docker.



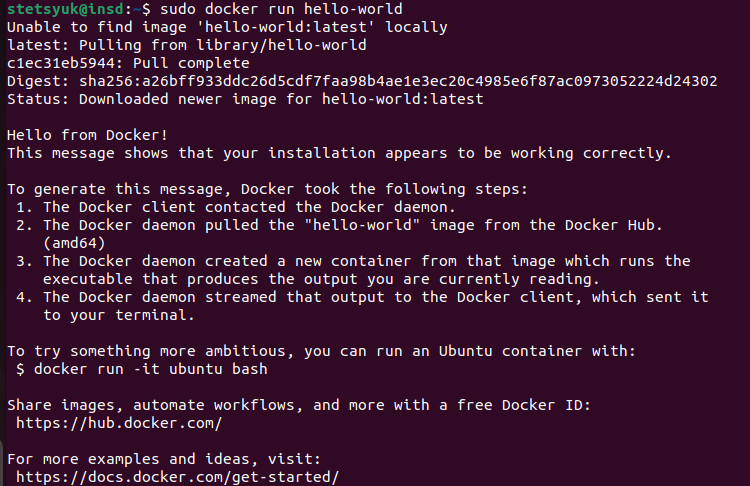


Далее можно приступить к установке.





Убедимся в правильности установки.



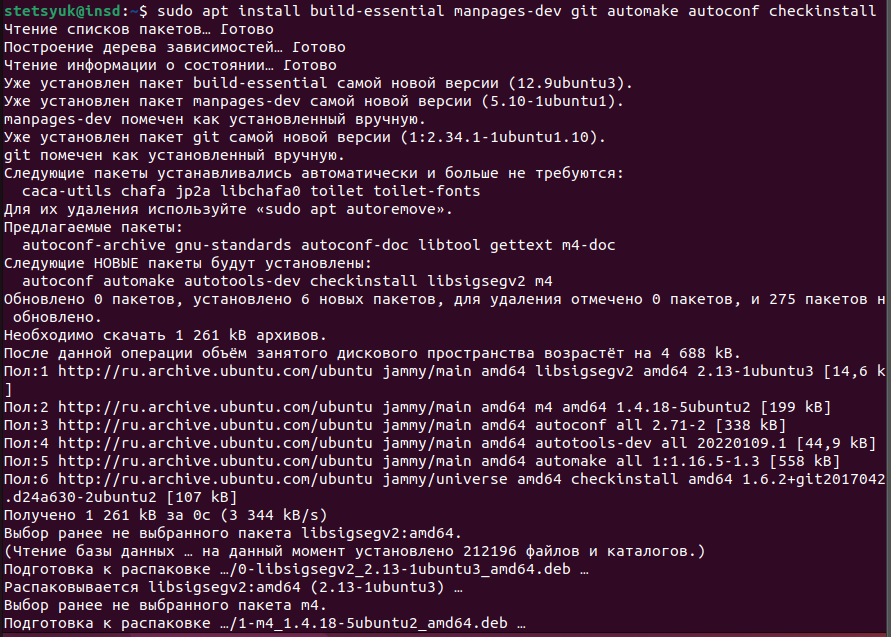
5 Практическая работа 9

**Задание**

Выполните сборку и установку одной из программ из исходных кодов согласно варианту:

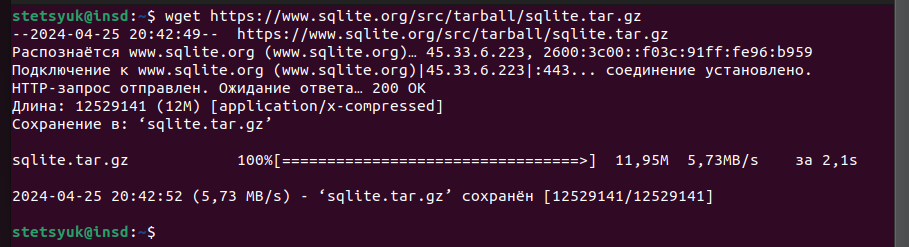
1. sqlite3;
2. gparted (необходимо брать версию для GNOME);
3. wget;
4. nano;
5. patch;
6. ncdu;
7. cowsay.

Изначально нужно установить компилятор.

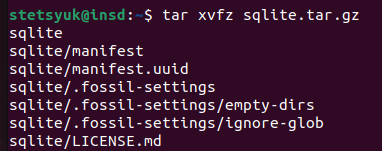




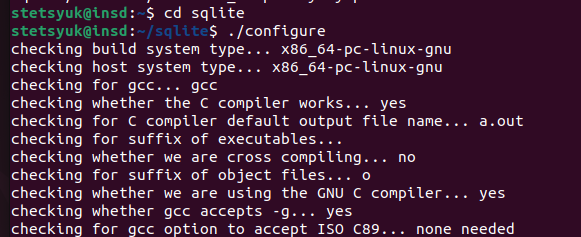
Далее необходимо скачать исходники sqlite3 в собственную директорию.



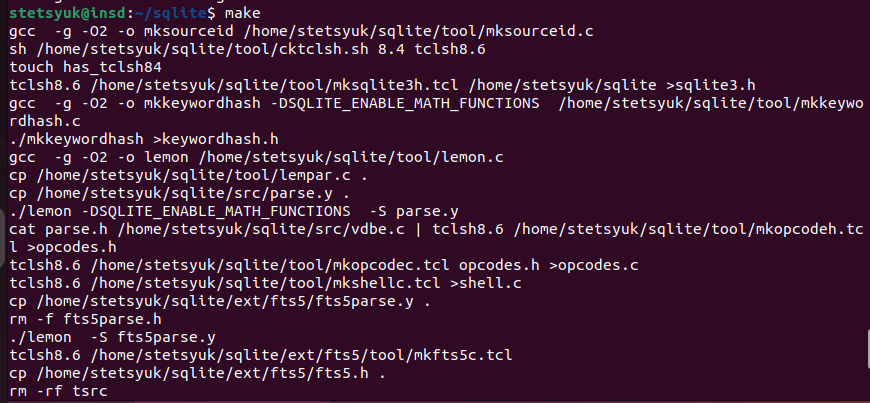
Теперь нужно распаковать при помощи команды tar xvfz.



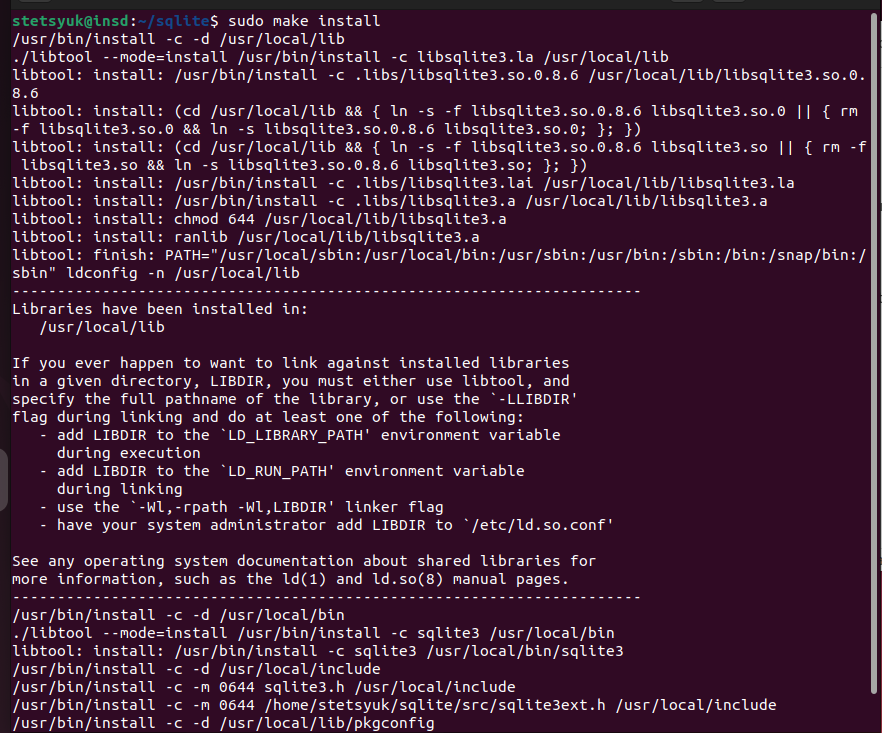
Распаковка прошла успешно и распакованный файл уже содержит скрипт configure.



После успешной установки зависимостей начнем сборку пакета.



Установим собранный пакет



Проверим корректность работы установленного приложения

