Министерство образования Республики Беларусь Учреждение образования

«Брестский государственный технический университет»

Лабораторная работа №3

По дисциплине «Операционные системы» Тема: «BASH: ПОТОКИ ДАННЫХ. ПРОГРАММИРОВАНИЕ»

Выполнил:

Студент 2 курса группы Ии-23

Вышинский Артём Сергеевич

Проверил:

Чеслов А. В.

Брест 2023

**Цель работы:** Получить практические и теоретические знания о системе Linux, потоках данных.

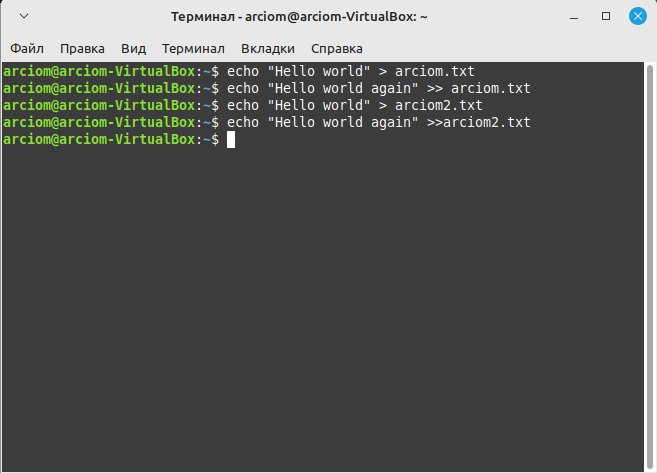
# Ход работы

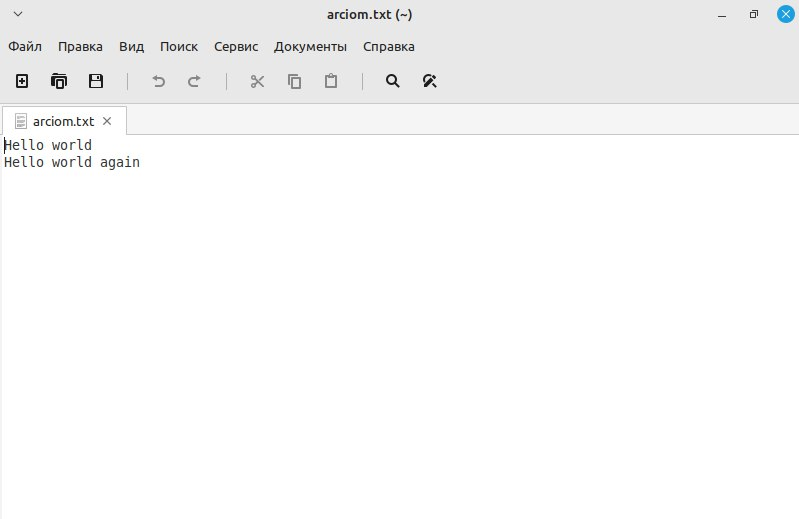
Часть 1

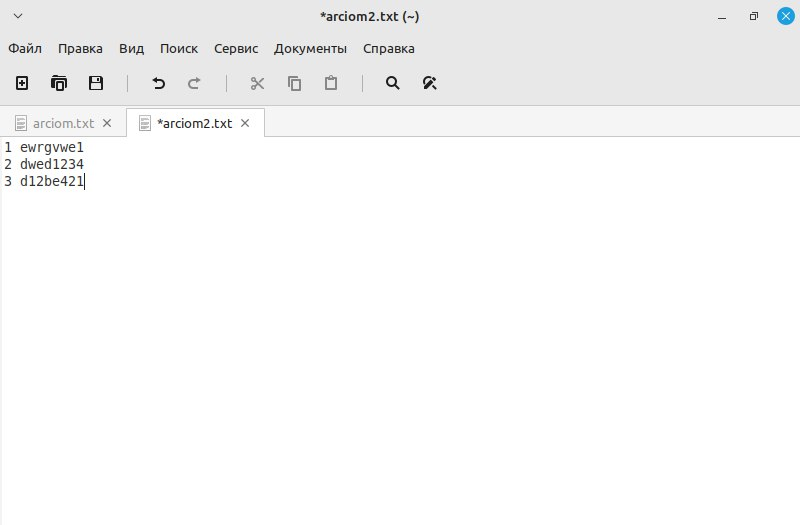
# Вывести любое сообщение с помощью команды echo перенаправив вывод:

* в несуществующий файл с помощью символа >;
* в несуществующий файл с помощью символа >>;
* в существующий файл с помощью символа >;

- в существующий файл с помощью символа >>;

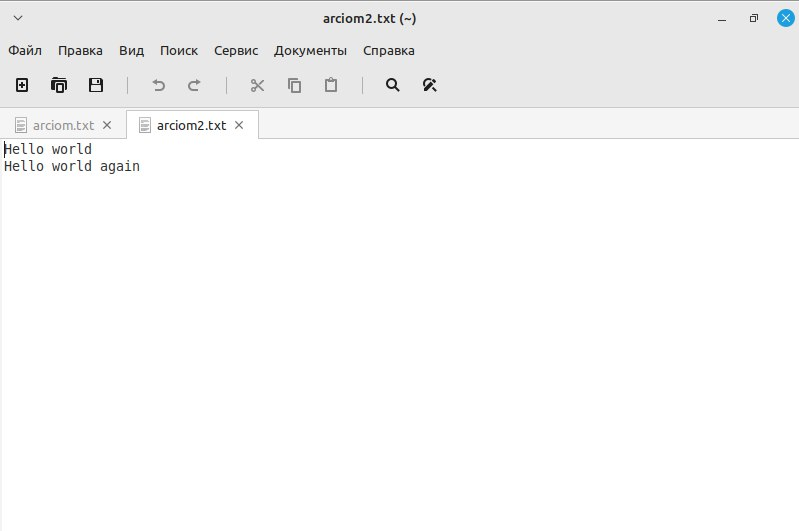






Перед командой:

Echo “Hello world” > arciom2.txt



После команд:

Echo “Hello world” > arciom2.txt

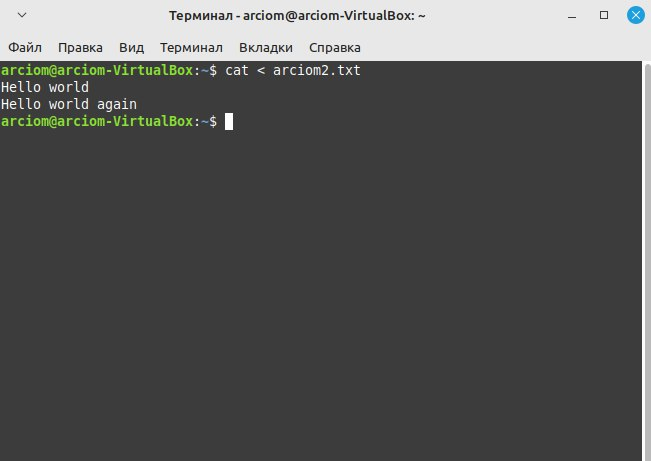
Echo “Hello world” >> arciom2.txt

Файл arciom2.txt был создан перед выполнением.

Объяснить результаты.

Текст в любом случае запишется, если файл не был создан до этого команда echo создаст и запишет текст в этот файл.

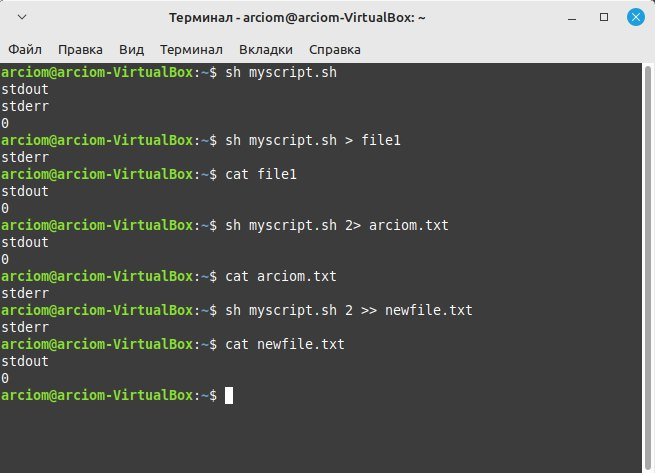
# Переадресовать стандартный ввод для команды cat на файл.



1. **Вывести сообщение с помощью команды echo в канал ошибок.**

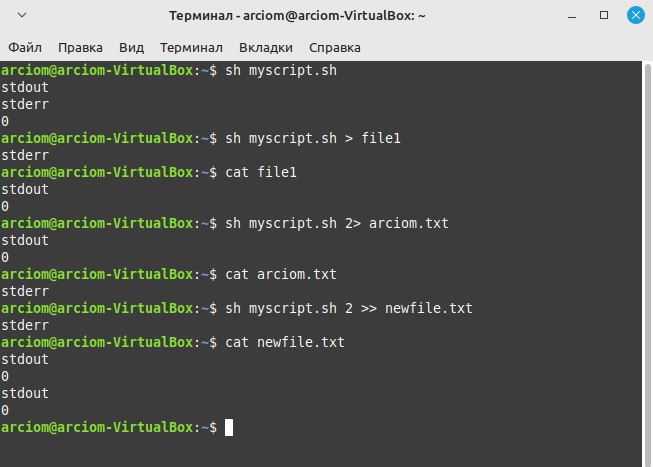
**файл myscript:**





# Запустить его:

**-без перенаправления (sh myscript);**

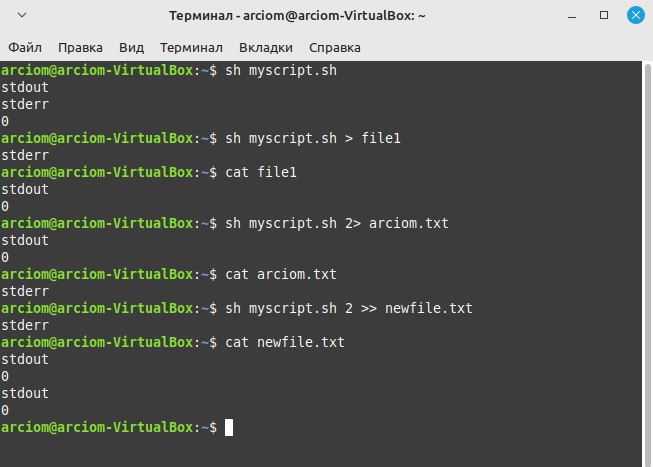


Выполняется файл myscript и команда echo в нем выводит stdout в стандартный вывод, и

stderr в канал ошибок.

# -перенаправив стандартный вывод в файл, просмотреть содержимое файла

**(sh myscript > file1);**



Т.к мы перенаправили стандартный вывод в файл в него записалось stdout, а stderr вывелось в стандартный канал ошибок.

# -перенаправить стандартный канал ошибок в существующий и несуществующий файлы с помощью символов > и >> ;

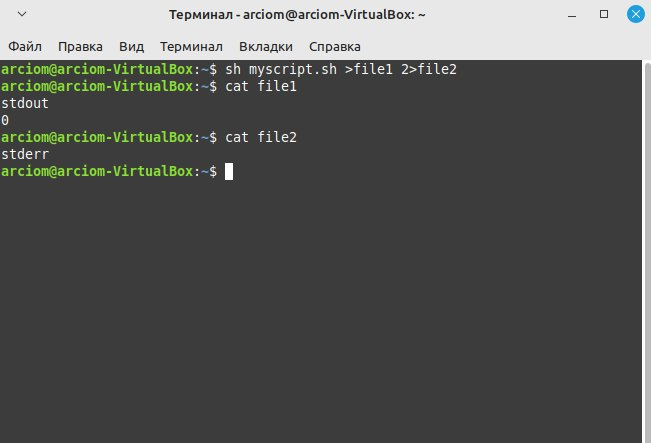
# 

Стандартный канал ошибок, переправляется в файлы, поэтому в файлах записывается stderr,

а во время выполнения стандартный вывод выводится stdout.

> - записывает в файл стирая предыдущие записи. >> - записывает в конец файла, не стирая предыдущие записи.

# -перенаправив стандартный вывод в файл 1, стандартный канал ошибок - в файл 2;



Перенаправив стандартный вывод в файл 1 в него запишется stdout, а в стандартный канал ошибок выведется stderr.

# - перенаправив стандартный вывод и стандартный канал ошибок в файл 3;

# 

Перенаправив стандартный вывод в файл 3 запишется stdout, а при перенаправлении стандартного канала ошибок в файл 3 в него записывается stder, при этом старая запись сотрется.

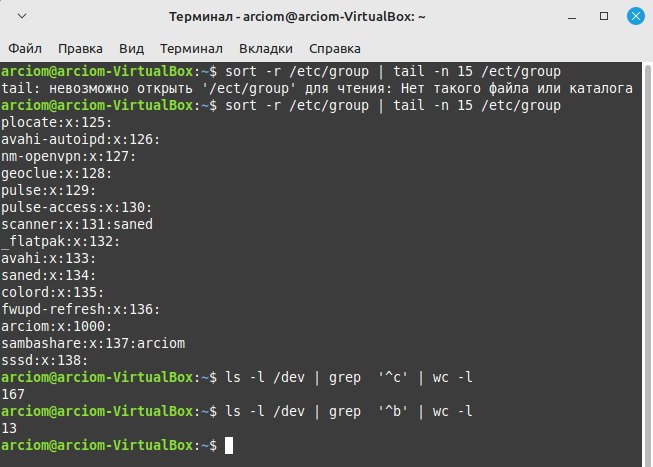
# - перенаправив стандартный вывод в файл 4 с помощью символа >,а стандартный канал ошибок в файл 4 с помощью символа >>;

# 

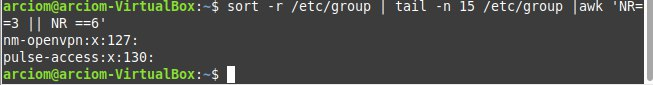
**Объяснить результаты.**

Происходит тоже самое, что и с файлом 3, только из-за того что мы используем >> stderr записывается в конец файла и при этом не стирается предыдущая запись.

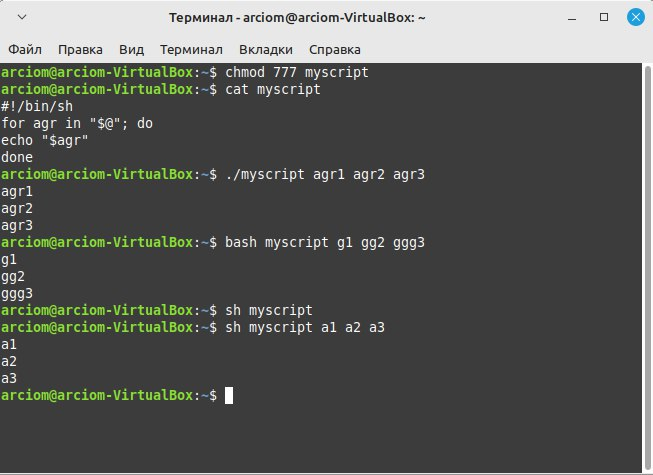
# Вывести третью и шестую строку из последних пятнадцати строк отсортированного в обратном порядке файла /etc/group.



1. **Подсчитать при помощи конвейера команд количество блочных и количество символьных устройств ввода-вывода, доступных в системе.**



# Написать скрипт, выводящий на консоль все аргументы командной строки, переданные данному скрипту. Привести различные варианты запуска данного скрипта, в том числе без непосредственного вызова интерпретатора в командной строке.



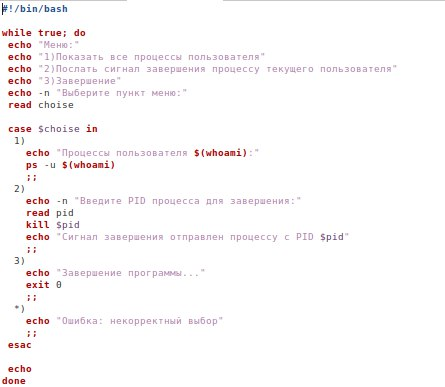
1. **Написать скрипт согласно индивидуальному заданию. Номер варианта согласовать с преподавателем.**

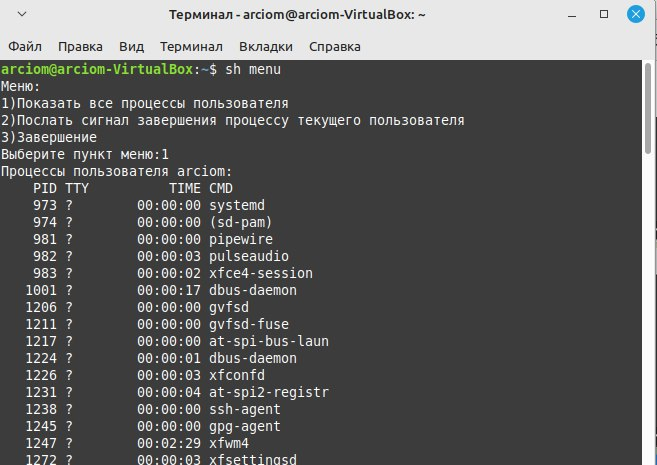
3. Реализовать командный файл, реализующий меню из трех пунктов (в цикле):

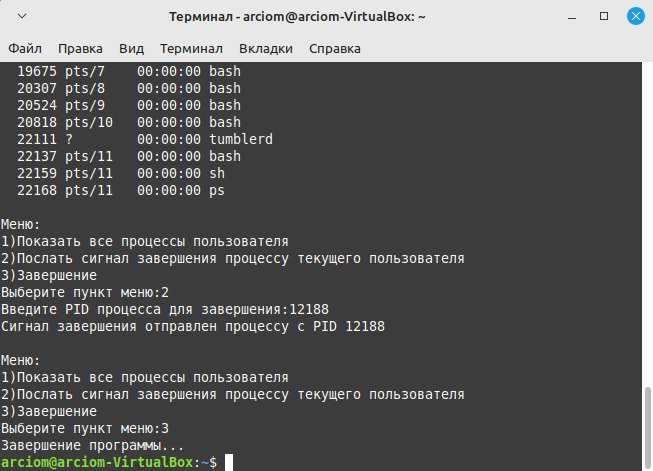
1) показать все процессы пользователя, запустившего данный командный файл;

2) послать сигнал завершения процессу текущего пользователя (ввести PID процесса);

1. завершение.







Вывод:Получил практические и теоретические знания о Linux,потоках данных.