

## Лабораторная работа №2.

**Задание 1:** Используя систему виртуализации (VirtualBox) установить ОС «ASTRA LINUX COMMON EDITION» (orel-current.iso). // <https://download.astralinux.ru/astra/stable/orel/iso/>

**Задание 2:** Добавить 3 пользователя в систему (student1, student2, Student3). Задать пароль на root – R\_o0T\_2020, student1 - StudenT\_2020, student2 - StudenT\_2021, Student3 - StudenT\_2022.

**Задание 3:** Создать папку в созданной виртуальной машине \home\common и разрешить ее только для чтения для student1, student2, Student3.

**Задание 4:** Разработать сценарий автоматизации на языке «Bash» согласно своему варианту (Приложение 1) и разместить ярлык для него на всех рабочих столах пользователей. *Обеспечить проверку на наличие необходимых по заданию параметров командной строки, возможности доступа к заданным пользователем каталогам или файлам. В случае возникновения исключений, сообщать об этом пользователям понятным для восприятия текстом на русском языке!*

**Задание 5:** Установить и настроить пакет обеспечения общего доступа к файловой системе (Samba) и сделать домашние каталоги пользователей student1, student2, Student3 доступными по сети, обеспечив полный доступ к ним. Каталог \home\common сделать доступным для чтения любыми сетевыми пользователями.

**Задание 6:** Используя планировщик и файл конфигурации «\etc\crontab», создать сценарий ежедневного (кроме выходных дней) удаления временных файлов (\*.bak; \*.tmp) во всех домашних каталогах пользователей в 12:00:00.

**Задание 7:** Используя входящую в комплект дистрибутива скрипт резервного копирования, обеспечить ежедневное резервирование данных пользователей.

**Задание 8:** Разработать сценарий автоматизации на языке «Perl» согласно своему варианту (Приложение 2) и разместить ярлык для него на всех рабочих столах пользователей. *Обеспечить проверку на наличие необходимых по заданию параметров командной строки, возможности доступа к заданным пользователем каталогам или файлам. В случае возникновения исключений, сообщать об этом пользователям понятным для восприятия текстом на русском языке!*

**Задание 9:** Создать отчет по работе, содержащий скриншоты выполнения заданий и тексты программ.

## Приложение 1 – Задания по bash

1. Написать скрипт для поиска файлов заданного размера в заданном каталоге (имя каталога задаётся пользователем в качестве третьего аргумента командной строки). Диапазон (min - max) размеров файлов задаётся пользователем в качестве первого и второго аргумента командной строки.
2. Написать скрипт, выводящий на консоль размеры и права доступа для всех файлов в заданном каталоге и всех его подкаталогах (имя каталога задается пользователем в качестве аргумента командной строки).
3. Написать скрипт для поиска заданной пользователем строки во всех файлах заданного каталога и всех его подкаталогах (строка и имя каталога задаются пользователем в качестве первого и второго аргумента командной строки). На консоль выводятся полный путь и имена файлов, в содержимом которых присутствует заданная строка, и их размер. Если к какому-либо каталогу нет доступа, необходимо вывести соответствующее сообщение и продолжить выполнение.
4. Написать скрипт поиска одинаковых по их содержимым файлов в двух каталогах, заданных пользователем в качестве первого и второго аргумента командной строки. В результате работы программы файлы, имеющиеся в Dir1, сравниваются с файлами в Dir2 по их содержимому. На экран выводятся число просмотренных файлов и результаты сравнения.
5. Написать скрипт, находящий в заданном каталоге и всех его подкаталогах все файлы, владельцем которых является заданный пользователь. Имя владельца и каталог задаются пользователем в качестве первого и второго аргумента командной строки. Скрипт выводит результаты в файл (третий аргумент командной строки) в виде "полный путь, имя файла, его размер". На консоль выводится общее число просмотренных файлов.
6. Написать скрипт, находящий в заданном каталоге и всех его подкаталогах все файлы заданного размера и принадлежащие определенному пользователю. Диапазон (min - max) размеров файлов задаётся пользователем в качестве первого и второго аргумента командной строки. Имя владельца и каталог задаются пользователем в качестве третьего и четвертого аргумента командной строки. Скрипт выводит результаты поиска в файл (пятый аргумент командной строки) в виде "полный путь, имя файла, его размер". На консоль выводится общее число просмотренных файлов.
7. Написать скрипт, подсчитывающий суммарный размер файлов в заданном пользователем каталоге и всех его подкаталогах (имя каталога задаётся пользователем в качестве первого аргумента командной строки). Скрипт выводит результаты подсчета в файл (второй аргумент командной строки) в виде "каталог (полный путь), суммарный размер файлов, число просмотренных файлов".
8. Написать скрипт синхронизации двух каталогов, заданных пользователем в качестве первого и второго аргумента командной строки. В результате работы программы файлы, имеющиеся в Dir1, но отсутствующие в Dir2, должны скопироваться в Dir2 вместе с правами доступа.
9. Написать скрипт поиска одинаковых по их содержимым файлов в двух каталогах, заданных пользователем в качестве первого и второго аргумента командной строки. В результате работы программы файлы, имеющиеся в Dir1, сравниваются с файлами в Dir2 по их содержимому.
10. Написать скрипт поиска заданной пользователем комбинации из m байт (первый аргумент командной строки.  $m < 255$ ) во всех файлах, заданных пользователем каталога в качестве второго аргумента командной строки.
11. Напишите скрипт, подсчитывающий количество скрытых файлов в домашнем каталоге заданного в командной строке пользователя. Поиск без захода в подкаталоги, скрытые каталоги не учитывать.

12. Напишите скрипт, получающий от пользователя в командной строке: имя каталога, расширение файлов и строку поиска. Скрипт должен найти все файлы с указанным расширением, содержащие указанную строку. Результат следует записать в файл `/var/log/строка.log`.

13. Напишите скрипт, подсчитывающий количество скрытых файлов в домашнем каталоге пользователя заданного в командной строке. Поиск без захода в подкаталоги, скрытые каталоги не учитывать.

14. Напишите скрипт, получающий от пользователя в командной строке имя каталога, расширение файлов и строку поиска. Скрипт должен найти все файлы с указанным расширением, не содержащие указанную строку символов. Результат следует записать в файл `/var/log/строка.log`.

15. Напишите скрипт, получающий от пользователя PID процесса. Скрипт должен определить название приложения, объем используемой памяти, владельца процесса и время его работы. Результат следует записать в файл `/var/log/процесс.log`.

16. Напишите скрипт, упаковывающий все файлы текущего каталога с одинаковыми расширениями в архив и размещающий этот архив в каталоге `/var/backups/текущая дата`. Поиск без захода в подкаталоги, скрытые файлы не учитывать. Например: для даты 01.02.2022 и каталога, в котором имеется 3 файла с расширением `txt` и пять файлов с расширением `doc`, в результате действия в скрипта в каталоге `/var/backups/20220201` должны появиться два новых архива. Один с именем `txt`, другой с именем `doc`.

17. Напишите скрипт, получающий от пользователя в командной строке: имя каталога и размер `N` в мегабайтах. Скрипт должен найти в указанном каталоге все файлы размера больше `N`, упаковать каждый из них в архив, удалить исходный файл и создать пустой файл с таким же именем. Отчет о выполненных скриптом действиях следует записать в файл `/var/log/фамилия.log`.

18. Напишите скрипт, получающий от пользователя в командной строке: имя каталога и размер `N` в мегабайтах. Скрипт должен найти в указанном каталоге все файлы размера больше `N`, упаковать каждый из них в архив, удалить исходный файл и создать пустой файл с таким же именем. Отчет о выполненных скриптом действиях следует записать в файл `/var/log/фамилия.log`.

19. Напишите скрипт, получающий от пользователя в командной строке: имя каталога, расширение файлов и строку поиска. Скрипт должен найти все файлы с указанным расширением, содержащие указанную строку. Результат следует записать в файл `/var/log/фамилия.log`.

20. Сгенерировать следующую информацию о процессах системы в файле `/var/log/процессы.log`, имеющих значение идентификатора, больше заданного пользователем в командной строке или сообщить об ошибке ввода идентификатора. Завершить выполнение всех процессов, владельцем которых является текущий пользователь.

21. Сгенерировать следующую информацию — полный листинг о процессах текущего пользователя в файле `/var/log/процессы.log`: PID, PPID, выделенное время ЦП, время запуска, размер образа. Приостановить выполнение процесса, владельцем которого является текущий пользователь.

22. Отобразить информацию о процессах указанного в командной строке пользователя в виде иерархии, вывод отсортировать по значениям PID в файле `/var/log/процессы.log`. приостановить выполнение процесса, владельцем которого является текущий пользователь.

23. Отобразить информацию о процессах, начиная с указанного в командной строке идентификатора, с выделением цветом текущего процесса и его предков. Завершить выполнение процесса, владельцем которого является текущий пользователь.

## Приложение 2 – Задания по Perl

1. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти файлы компоненты которых являются действительными числами. Найти наибольшее из значений модулей этих компонентов с указанием полного пути к файлу где он находится (таких файлов может быть несколько в случае одинаковых данных).

2. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти файлы компоненты которых являются действительными числами. Найти сумму наибольшего и наименьшего из значений компонент с указанием полного пути к файлу где он находится (таких файлов может быть несколько в случае одинаковых данных).

3. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти файлы компоненты которых являются действительными числами. Найти порядковый номер минимального числа в каждом из файлов, а также минимальное число из всех найденных файлов с указанием полного пути к файлу где он находится (таких файлов может быть несколько в случае одинаковых данных).

4. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти файлы компоненты которых являются действительными числами. Найти сумму чисел первого и последнего из найденных файлов (если указанных файл только один или нет, то сообщить об этом пользователю).

5. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти файлы компоненты которых являются действительными числами. Найти среднее арифметическое всех положительных чисел найденных файлов (в случае отсутствия указанных файлов или отсутствия в них положительных чисел, сообщить об этом пользователю).

6. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти файлы компоненты которых являются целыми числами. Найти самое последнее отрицательное число в найденных файлах с указанием полного пути к файлу где оно находится. (Если такого числа нет, то сообщить об этом пользователю).

7. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти подсчитать число вхождений в файлы заданных пользователем вторым параметром командной строки сочетаний символов с указанием полного пути к файлу где оно находится в каждом из найденных файлов.

8. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти все текстовые файлы. Все четные строки этих файлов записать в файл /var/log/четные.txt, а нечетные в файл /var/log/нечетные.txt. (Проверить доступность на запись файлов и сообщить пользователю в случае невозможности этих операций).

9. В заданном пользователем в командной строке каталоге, найти все текстовые файлы с одинаковым количеством строк с указанием полного пути к этим файлам (Файлы могут располагаться в разных подкаталогах).

10. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти все текстовые файлы. Удалить, в найденных файлах, лишние пробелы и знаки табуляции. (в случае невозможности изменения файлов, сообщить об этом пользователю).

11. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти все текстовые файлы. Добавить, в найденных файлах, в начало каждой строки по одному пробелу (в случае невозможности изменения файлов, сообщить об этом пользователю).

12. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти все текстовые файлы. Дописать в конце найденных файлов

следующие данные: количество строк, количество символов в каждой строке (в случае невозможности изменения файлов, сообщить об этом пользователю).

13. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти все текстовые файлы. Удалить, в найденных файлах, третью строку (если она имеется, если нет сообщить об этом пользователю. В случае невозможности изменения файлов, также сообщить об этом пользователю).

14. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти все текстовые файлы, в которых количество цифровых данных превышает количество текстовой информации (пробелы не учитывать).

15. В заданном пользователем в командной строке каталоге (число подкаталогов неограниченно), найти все текстовые файлы и выровнять в них ширину строк, используя пробелы (в случае невозможности изменения файлов, сообщить об этом пользователю). Сообщить о количестве обработанных файлов и общего времени их форматирования.