### Преподаватель - Сазонова Полина Андреевна

[psazonova@gmail.com](mailto:psazonova@gmail.com)

<https://vk.com/sazonova_polina>

### **Курс информатики, семинар #1**

Полезные команды линуксовой консоли:

* vim, nano, emacs - текстовые редакторы
* man - справка по введенной теме
* cat - передает вход в выход или печатает содержимое файла на экран
* grep - фильтрует вход и передает его в выход
* find - ищет файл на файловой системе
* echo - выводит строку в выходной поток
* sh, bash, zsh, eval - интерпретаторы команд
* ps - показывает запущенные процессы
* touch - изменяет время модификации файла на текущее или создает пустой файл
* ls - выводит содержимое директории
* less, more - удобные просмотрщики содержимого файлов
* cd - изменяет текущую директорию
* mkdir - создает директорию
* rm - удаляет файл или директорию
* pwd - печатает полное имя текущей директорию
* wc - выводит различную статистику, например количество строк во входе
* expr - осуществляет арифметические операции
* tee - распараллеливает вход

Если что-то непонятно — используйте ***man <команда>*** (от слова manual, справочник) или идите в Google.

Очень хорошая книга по Unix shell: <http://www.grymoire.com/Unix/index.html>

##### **1 часть, уровень Novice**

Команды cd, ls, touch, cp, mv, rm, mkdir, mcedit, pwd.

Вывод файла на экран: cat, less

Прерывание программы по сигналу: CTRL+С

Задания:

1. Создайте директорию, создайте файл, поперемещайте и покопируйте их. Попытайтесь удалить созданную директорию.

2. Каков полный путь до вашей домашней директории?

3. Что делает команда cd без параметров?

4. Чем отличается cat от less?

##### **2 часть, уровень Advanced**

Команды find и grep

Перенаправление вывода: >, >>, |. stdin и stderr

Задания:

0. Скачайте крупный текстовый файл, выполнив команду **wget http://factorized.net/crusoe.txt**

1. Найти все файлы \*.txt в папке /tmp

2. Найти все строки "friday" в большом текстовом файле

3. С помощью > и >> записать в текстовый файл фразу "Hello world"

4. Вывести в файл список файлов в текущей директории

5. Попробовать найти все файлы \*.txt на всём диске. Придумать, как уменьшить количество мусора, выводимого на экран этой командой.

##### **3 часть, уровень Expert**

Более умная обработка файлов: wc, awk, sed

Полезно почитать: man hier (от слова hierarchy, иерархия)

Текстовый редактор для простых смертных: mcedit

Задания:

1. Найти количество упоминаний слова “Friday” в большом текстовом файле.

2. Заменить в текстовом файле все слова “Friday” на “Saturday”

3. Свободное задание на арифметические действия с данными из файла с помощью awk. Возможны следующие варианты заданий:

* Придумать задание самостоятельно, согласовать его с преподавателем, решить и сдать.
* Подсчитать сумму чисел по столбцам   
    
  Дан входной файл, где на каждой строке находятся два целых числа, разделенных пробелом. Надо посчитать сумму чисел в первом столбце и во втором. Вывести результат на экран.

|  |  |
| --- | --- |
| Входной файл | Ожидаемый вывод |
| 1 2  2 2  3 7 | 6 11 |

* Вывести строку, содержащую больше всего чисел.  
  Дан входной файл, в котором на каждой строке находится несколько целых чисел, разделенных пробелом. Надо вывести строку, в которой чисел больше всего. Если строк с одинаковым количеством чисел несколько, то вывести первую. Кроме чисел, разделенных пробелами в файле ничего нет.

|  |  |
| --- | --- |
| Входной файл | Ожидаемый вывод |
| 1 300  2 3  10 | 1 300 |
| 200  1 1 1 1  10 10 10 | 1 1 1 1 |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### **Курс информатики, семинар #2**

Очень хорошая книга по Unix shell: <http://www.grymoire.com/Unix/index.html>

##### **1 часть, уровень Novice**

Скрытые файлы (hidden files)

Команда ls -a

Задания:

1. Создайте файл, имя которого начинается с точки. Убедитесь, что его не видно с помощью обычной команды **ls**

2. Выясните, какие скрытые файлы и директории находятся в вашей домашней директории

3. Ищет ли find по скрытым файлам и директориям?

##### **2 часть, уровень Advanced**

Команды which и whereis

Переменная окружения $PATH

Запуск исполняемых файлов из текущей директории и из произвольной директории.

Задания:

1. С помощью команд **which** и **whereis** выясните, в какой директории находятся программы cat, less, grep и find

2. Выведите на экран значение переменной окружения $PATH (***echo $PATH***)и убедитесь, что директория, найденная в п.1 входит в $PATH

3. Скомпилируйте какую-нибудь программу на C и попробуйте запустить её, находясь в той же директории. Перейдите в другую директорию и запустите ту же программу.

4. Добавьте директорию, в которой лежит программа из п.3 в $PATH (***export PATH=$PATH:<новая директория>***).

##### **3 часть, уровень Expert**

Пользователи и группы в UNIX.

Права доступа к файлам и директориям

Команды chmod и chown

Задания:

1. Создайте в своей домашней директорию файл под названием shared.txt. Объясните, какими правами доступа он обладает (для этого можно воспользоваться командой **ls -la**)

2. C помощью команды **chmod** сделайте так, чтобы созданный файл был доступен только его владельцу и только на чтение.

3. С помощью команды **chown** сделайте вашего соседа справа владельцем файла shared.txt, затем верните соседу слева полученный от него файл.

4. Проделайте те же операции, что и в п.1-3, но с директорией

5. Объясните что было бы, если операции из пункта 3 и 4 можно было бы делать обычным пользователям. Какие проблемы безопасности это имеет и почему они были запрещены.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

### **Курс информатики, семинар #3**

##### **1 часть, уровень Novice**

DNS и роутинг: команды ping, traceroute, dig и nslookup

Задания:

1. Объясните, что делают команды ping и traceroute. Приведите пример их использования.

2. Какой IP-адрес у сервера [yandex.ru?](http://yandex.ru?) Какой командой вы это определили и почему решили использовать именно ее?

##### **2 часть, уровень Advanced**

Bash-скрипты

Задания:

1. Напишите скрипт, принимающий в качестве параметра командной строки число и выводящий на экран какую-нибудь строку указанное число раз

$ ./write.sh 3

Hello

Hello

Hello

2. Напишите скрипт, суммирующий два числа, переданных в качестве параметров командной строки

$ ./add.sh 2 3

5

$ ./add.sh 10 15

25

3. Усовершенствуйте скрипт из предыдущего пункта так, чтобы в качестве третьего параметра можно было передавать желаемое арифметическое действие над числами

$ ./op 2 3 add

5

$ ./op.sh 10 7 sub

3

$ ./op.sh 2 4 mul

8

4. Усовершенствуйте скрипт из предыдущего пункта так, чтобы при отсутствии третьего параметра он брал необходимую операцию из переменной окружения ARITHMETIC\_OP

$ export ARITHMETIC\_OP=add

$ ./op.sh 3 4

7

$ export ARITHMETIC\_OP=mul

$ ./op.sh 3 4

12

$ ./op.sh 3 4 add

7

##### **3 часть, уровень Expert**

Регулярные выражения (regular expressions).

Задания:

1. С помощью egrep и регулярных выражений найдите в файле http://factorized.net/crusoe.txt:

* все слова, начинающиеся на 'z'
* все слова из 16 букв
* все слова, начинающиеся на 'a' и заканчивающиеся на 'c'
* все слова, начинающиеся на 'ab', но у которых третья буква не 'o'

Словом считать любую последовательность символов ограниченную с обоих сторон пробельными символами, либо следующими знаками препинания: точка, запятая, точка с запятой, вопросительный и восклицательный знаки, двойная и одинарная кавычки

2. Найдите в файле http://factorized.net/patterns.txt строки, которые:

* состоят только из цифр
* состоят только из букв

3. Как вы знаете, в российских автомобильных номерах могут использоваться только буквы, имеющие латинские графические аналоги: А, В, Е, К, М, Н, О, Р, С, Т, У и Х.

Найдите в файле из п.2 все строки, которые могут являться валидными российскими автомобильными номерами (используя заглавные английские буквы).

Валидным номером будем считать следующую комбинацию:

* буква
* три цифры
* две буквы
* две или три цифры

Например, В532OT154, или X151BT00 или Y010MK999