

# Презентация по лабораторной работе №10

Операционные системы

---

Сячинова Ксения Ивановна

22 марта 2023

Российский университет дружбы народов, Москва, Россия

## Цель работы

---

Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы

## Задание

---

1. Написать скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в вашем домашнем каталоге. При этом файл должен архивироваться одним из архиваторов на выбор zip, bzip2 или tar. Способ использования команд архивации необходимо узнать, изучив справку.
2. Написать пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов.

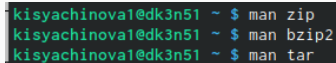
3. Написать командный файл — аналог команды `ls` (без использования самой этой команды и команды `dir`). Требуется, чтобы он выдавал информацию о нужном каталоге и выводил информацию о возможностях доступа к файлам этого каталога.
4. Написать командный файл, который получает в качестве аргумента командной строки формат файла (`.txt`, `.doc`, `.jpg`, `.pdf` и т.д.) и вычисляет количество таких файлов в указанной директории. Путь к директории также передаётся в виде аргумента командной строки.

## Выполнение лабораторной работы

---

1. Создадим папку “lab10”, где будут все наши скрипты.

Для начала изучим команды для архивации. Для этого используем команду “man”.

A terminal window with a dark background and green text. It shows three lines of commands being executed by a user named kisyachinova1 on a machine named dk3n51. The commands are 'man zip', 'man bzip2', and 'man tar'.

```
kisyachinova1@dk3n51 ~ $ man zip  
kisyachinova1@dk3n51 ~ $ man bzip2  
kisyachinova1@dk3n51 ~ $ man tar
```

Рис. 1: Команды



```

ZIP(1L)                                ZIP(1L)

NAME
    zip - package and compress (archive) files

SYNOPSIS
    zip [-aABcdDffGhJkLmqR$TuvVwXyz185] [--longoption ...] [-b path] [-n
    suffixes] [-t date] [-tt date] [zipfile [file ...]] [-x list]

    zipcloak (see separate man page)

    zipnote (see separate man page)

    zipsplit (see separate man page)

Note: Command line processing in zip has been changed to support long op-
tions and handle all options and arguments more consistently. Some old
command lines that depend on command line inconsistencies may no longer

```

Рис. 2: zip

```

LBZIP2(1)                                User commands                                LBZIP2(1)

NAME
    lbzip2 - parallel bzip2 utility

SYNOPSIS
    lbzip2|bzip2 [-n NUM] [-k|-c|-t] [-d] [-1 .. -9] [-f] [-s] [-u] [-v] [-s]
    [ FILE ... ]

    lbunzip2|bunzip2 [-n NUM] [-k|-c|-t] [-z] [-f] [-s] [-u] [-v] [-S] [ FILE
    ... ]

    lbzcat|bzcat [-n NUM] [-z] [-f] [-s] [-u] [-v] [-S] [ FILE ... ]

    lbzip2|bzip2|lbunzip2|bunzip2|lbzcat|bzcat -h

DESCRIPTION
    Compress or decompress FILE operands or standard input to regular files or

```

Рис. 3: bzip2

```
TAR(1)                                GNU TAR Manual                                TAR(1)

NAME
    tar - an archiving utility

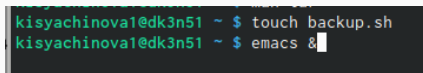
SYNOPSIS
    Traditional usage
    tar {A|c|d|r|t|u|x}[GnSkUWoMpsMBiajJzZhPlRvwo] [ARG...]

    UNIX-style usage
    tar -A [OPTIONS] ARCHIVE ARCHIVE

    tar -c [-f ARCHIVE] [OPTIONS] [FILE...]
    tar -d [-f ARCHIVE] [OPTIONS] [FILE...]
    tar -t [-f ARCHIVE] [OPTIONS] [MEMBER...]
```

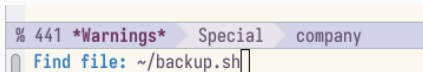
Рис. 4: tar

Далее создадим файл, в котором будет написан скрипт. Откроем его с помощью редактора “emacs” (сочетание клавиш “с+х”, “с+f”).

A terminal window with a dark background. The prompt is 'kisyachinova1@dk3n51 ~ \$'. The first command is 'touch backup.sh' and the second is 'emacs &'.

```
kisyachinova1@dk3n51 ~ $ touch backup.sh
kisyachinova1@dk3n51 ~ $ emacs &
```

Рис. 5: Создание файла

A file manager window with a light yellow background. The top bar shows '% 441 \*Warnings\* Special company'. The main area shows 'Find file: ~/backup.sh' with a cursor at the end of the text.

```
% 441 *Warnings* Special company
Find file: ~/backup.sh
```

Рис. 6: Открытие файла

Напишем скрипт согласно задания. При написании скрипта я буду использовать архиватор “bzip2”.

```
#!/bin/bash

name='backup.sh'      # Сохранение файла со скриптов в переменную "name"
mkdir ~/backup         # Создание каталога ~/backup
bzip2 -k ${name}       # Архивирование скрипта
mv ${name}.bz2 ~/backup/ # Перемещение архивированного скрипта
echo "Выполнено"
```

Рис. 7: Текст скрипта

Сохраняем файл. Добавим право на выполнение с помощью команды “chmod +x \*.sh” и проверим работу скрипта. Также проверим, появился ли каталог и просматриваем его содержимое, где видим наш архив. Для просмотра архива введём команду “bunzip2 -c backup.sh.bz2”

```
kisyachinova1@dk3n51 ~ $ ls
backup.sh  GNUstep  tmp      Документы  Музыка  Шаблоны
backup.sh~ lab07.sh~ work     Загрузки  Общедоступные
bin        public    Видео    Изображения  'Рабочий стол'
[1]*  Завершён      emacs
kisyachinova1@dk3n51 ~ $ chmod +x *.sh
kisyachinova1@dk3n51 ~ $ ./backup.sh
Выполнено
kisyachinova1@dk3n51 ~ $ cd backup
kisyachinova1@dk3n51 ~/backup $ ls
backup.sh  bz2
kisyachinova1@dk3n51 ~/backup $ bunzip2 -c backup.sh.bz2
#!/bin/bash

name='backup.sh'          # Сохранение файла со скриптов в переменную "name"
mkdir ~/backup            # Создание каталога ~/Backup
bzip2 -k ${name}          # Архивирование скрипта
mv ${name}.bz2 ~/backup/  # Перемещение архивированного скрипта
echo "Выполнено"
kisyachinova1@dk3n51 ~/backup $
```

Рис. 8: Проверка действий

2. Создадим файл для второго скрипта и откроем его.

A terminal window with a black background and green text. It shows two commands being executed in sequence. The first command creates a file named 'os2.sh' in the current directory. The second command opens the file in the 'emacs' text editor and runs it in the background.

```
kisyachinova1@dk3n51 ~/work/os/lab10 $ touch os2.sh  
kisyachinova1@dk3n51 ~/work/os/lab10 $ emacs &
```

Рис. 9: Создание файла

Напишем соответствующий скрипт.

```
#!/bin/bash  
  
echo "Аргументы"  
for a in $@      # Цикл для перехода по введённым аргументам  
do echo $a      # Вывод аргумента  
done
```

Рис. 10: Текст скрипта

Проверим работу скрипта. Для это используем команду “./os2.sh”Аргументы””. Но для начала дадим право на выполнение. Введём количество аргументов больше и меньше 10. Скрип работает корректно.

```
kisyachinova1@dk3n51 ~/work/os/lab10 $ chmod +x *.sh
kisyachinova1@dk3n51 ~/work/os/lab10 $ ls
backup  os2.sh  os2.sh~
kisyachinova1@dk3n51 ~/work/os/lab10 $ ./os2.sh 0 1 2 3 4
Аргументы
0 1 2 3 4
0 1 2 3 4
0 1 2 3 4
0 1 2 3 4
0 1 2 3 4
kisyachinova1@dk3n51 ~/work/os/lab10 $ ./os2.sh 0 1 2 3 4 5 6 7 8 9 10 11
Аргументы
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
0 1 2 3 4 5 6 7 8 9 10 11
```

Рис. 11: Проверка скрипта



3. Создадим файл для написания третьего скрипта. Также открываем его в “emacs”.

A terminal window with a black background and green text. It shows two commands being executed in a shell. The first command is 'touch os3.sh' and the second is 'emacs \$'. The prompt for both is 'kisyachinova1@edk3n51 ~/work/os/lab10 \$'.

```
kisyachinova1@edk3n51 ~/work/os/lab10 $ touch os3.sh
kisyachinova1@edk3n51 ~/work/os/lab10 $ emacs $
```

Рис. 12: Создание файла

Напишем командный файл — аналог команды `ls` (без использования самой этой команды и команды `dir`).

```
#!/bin/bash
a="$1"
for i in ${a}/*
do
    echo "$1"

    if test -f $i
    then echo "Обычный файл"
    fi

    if test -d $i
    then echo "Каталог"
    fi

    if test -r $i
    then echo "Чтение разрешено"
    fi

    if test -w $i
    then echo "Запись разрешена"
    fi

    if test -x $i
    then echo "Выполнение разрешено"
    fi
done
```

Рис. 13: Текст скрипта

Затем даём право на выполнение и проверяем работу скрипта.

```
kisyachinova1@dk3n51 ~/work/os/lab10 $ chmod +x *.sh
kisyachinova1@dk3n51 ~/work/os/lab10 $ ls
backup  os2.sh  os2.sh~  os3.sh  os3.sh~
kisyachinova1@dk3n51 ~/work/os/lab10 $ ./os3.sh
```

```
Каталог
Чтение разрешено
Выполнение разрешено
```

```
Каталог
Чтение разрешено
Выполнение разрешено
```

```
Каталог
Чтение разрешено
Выполнение разрешено
```

```
Каталог
Чтение разрешено
Выполнение разрешено
```

Рис. 14: Проверка

4. Для выполнения последнего пункта создаём файл для работу и открываем его.

```
kisyachinova1@dk2n26 ~/work/os/lab10 $ touch os4.sh  
kisyachinova1@dk2n26 ~/work/os/lab10 $ emacs $
```

Рис. 15: Создание файла

Напишем скрипт по заданию.

```
#!/bin/bash
b="$1"
shift
for a in $@
do
    k=0
    for i in ${b}/*.${a}
    do
        if test -f "$i"
        then
            let k=k+1
        fi
    done
    echo "$k файлов содержится в $b с расширением $a"
done
```

Рис. 16: Текст скрипта

Далее проверим работу. Дадим право на исполнение и в домашнем каталоги создадим пару файлов. Првоерим работу.

```
kisyachinova1@dk2n26 ~ $ touch os4.pdf os4.doc os44.doc
kisyachinova1@dk2n26 ~ $ cd work
kisyachinova1@dk2n26 ~/work $ cd os
kisyachinova1@dk2n26 ~/work/os $ cd lab10
kisyachinova1@dk2n26 ~/work/os/lab10 $ ./os4.sh ~ pdf sh txt doc
1 файлов содержится в /afs/.dk.sci.pfu.edu.ru/home/k/i/kisyachinova1 с расширением pdf
0 файлов содержится в /afs/.dk.sci.pfu.edu.ru/home/k/i/kisyachinova1 с расширением sh
0 файлов содержится в /afs/.dk.sci.pfu.edu.ru/home/k/i/kisyachinova1 с расширением txt
2 файлов содержится в /afs/.dk.sci.pfu.edu.ru/home/k/i/kisyachinova1 с расширением doc
```

Рис. 17: Проверка

## Выводы

---

В процессе выполнения данной лабораторной работы я изучила основы программирования в оболочке ОС UNIX/Linux, научилась писать небольшие командные файлы