

Программирование в командном процессоре ОС UNIX. Командные файлы

Лабораторная работа No 10.

Козлова Нонна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	12
	Список литературы	13

Список иллюстраций

3.1	Пользуюсь командами touch и chmod	7
3.2	работает	8
3.3	работает	9
3.4	работает	10

Список таблиц

1 Цель работы

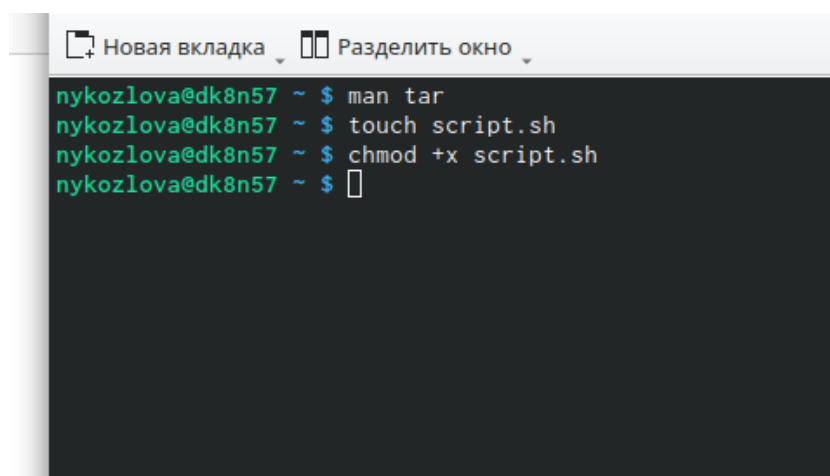
Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.

2 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.

3 Выполнение лабораторной работы

1. Создаю файл script и делаю его исполняемым (рис. 3.1).



```
nykozlova@dk8n57 ~ $ man tar
nykozlova@dk8n57 ~ $ touch script.sh
nykozlova@dk8n57 ~ $ chmod +x script.sh
nykozlova@dk8n57 ~ $
```

Рис. 3.1: Пользуюсь командами touch и chmod

2. Пишу скрипт, который при запуске будет делать резервную копию самого себя в другую директорию backup в моем домашнем каталоге. (рис. ??).

```
script.sh - GNU Emacs at dk8n57
File Edit Options Buffers Tools Sh-Script Help
New File Open Open Directory Close Save Undo Cut Copy Paste S
14 #!/bin/bash
mkdir ~/backup
cp script.sh ~/backup/backup.sh
gzip ~/backup/backup.sh
```

3. Запускаю файл (рис. ??).

```
nykozlova@dk8n57 ~ $
nykozlova@dk8n57 ~ $ ./script.sh
nykozlova@dk8n57 ~ $
```

4. Проверяю, что он сработал (рис. 3.2).

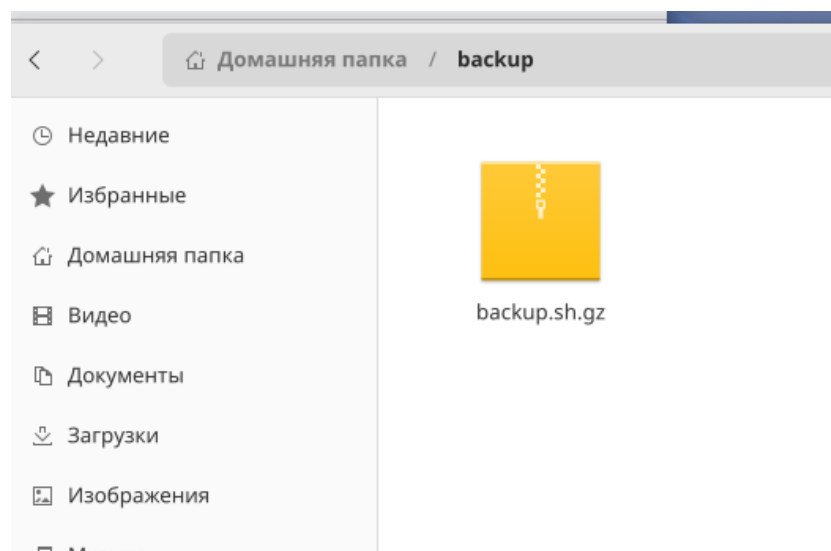
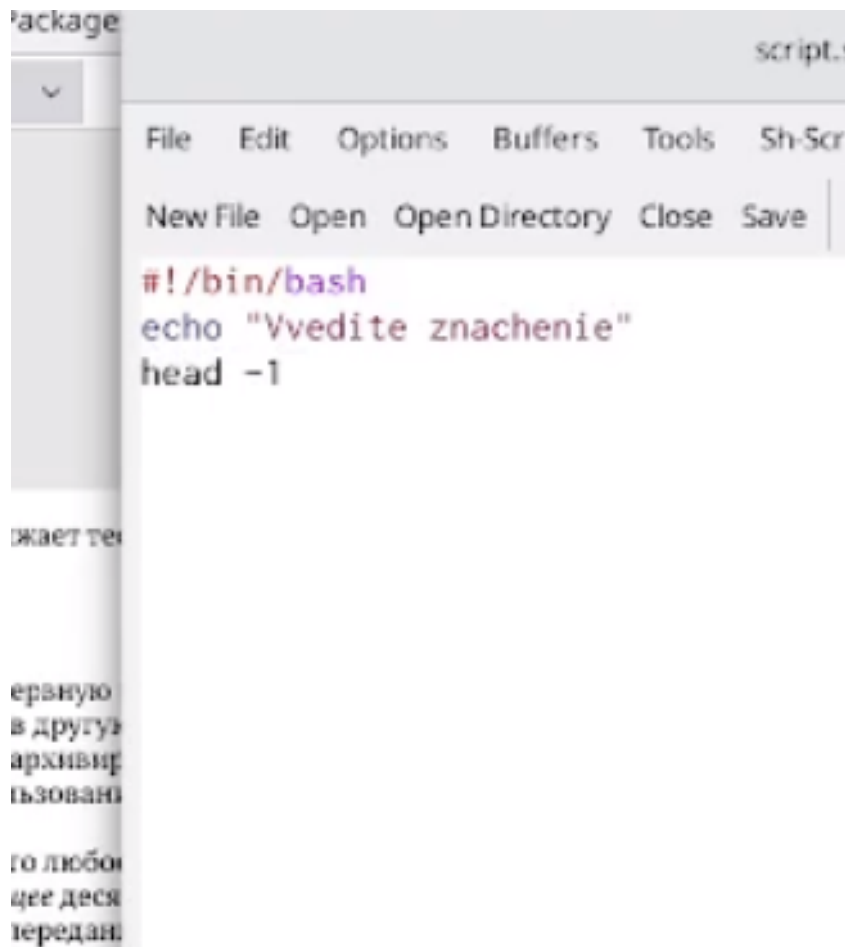


Рис. 3.2: работает

5. Пишу скрипт, который может последовательно распечатывать значения всех переданных аргументов (рис. ??).



6. Проверяю работу (рис. 3.3).

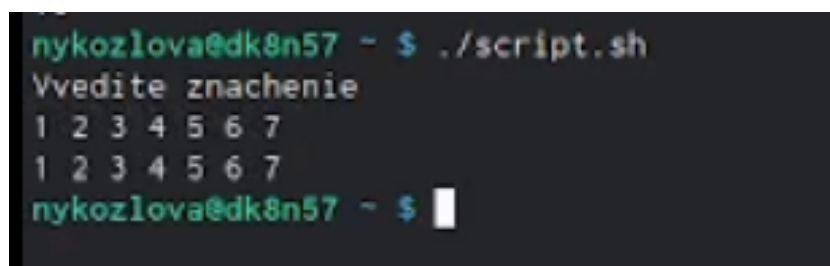
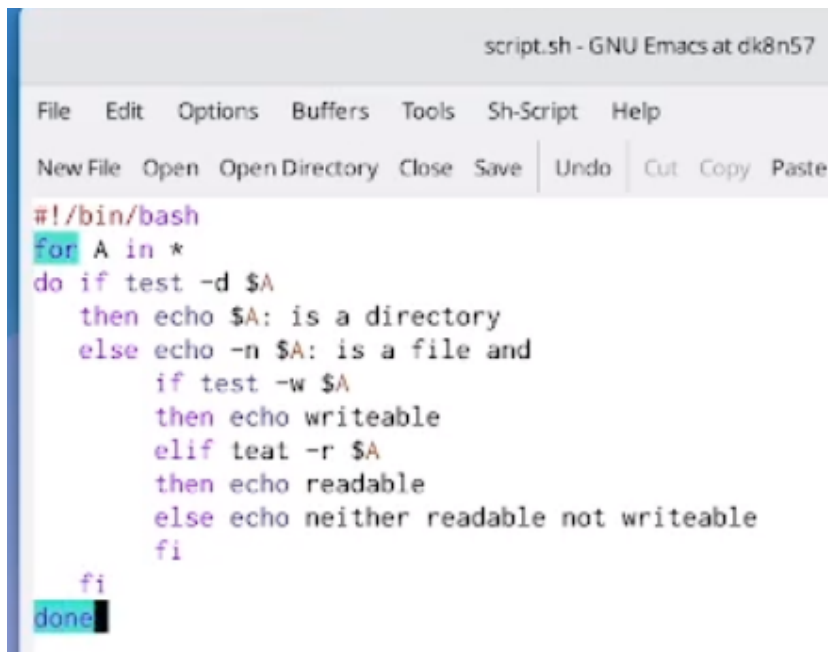


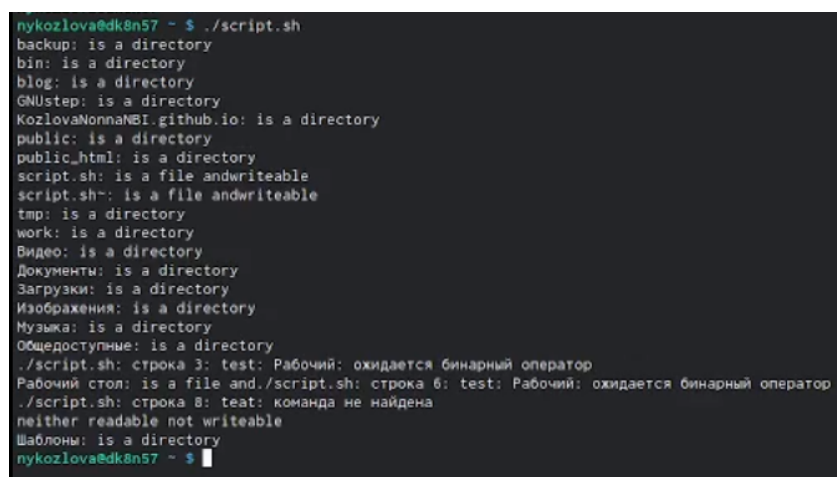
Рис. 3.3: работает

7. Пишу командный файл — аналог команды ls (рис. ??).



```
script.sh - GNU Emacs at dk8n57
File Edit Options Buffers Tools Sh-Script Help
NewFile Open OpenDirectory Close Save Undo Cut Copy Paste
#!/bin/bash
for A in *
do if test -d $A
then echo $A: is a directory
else echo -n $A: is a file and
if test -w $A
then echo writeable
elif teat -r $A
then echo readable
else echo neither readable not writeable
fi
fi
done
```

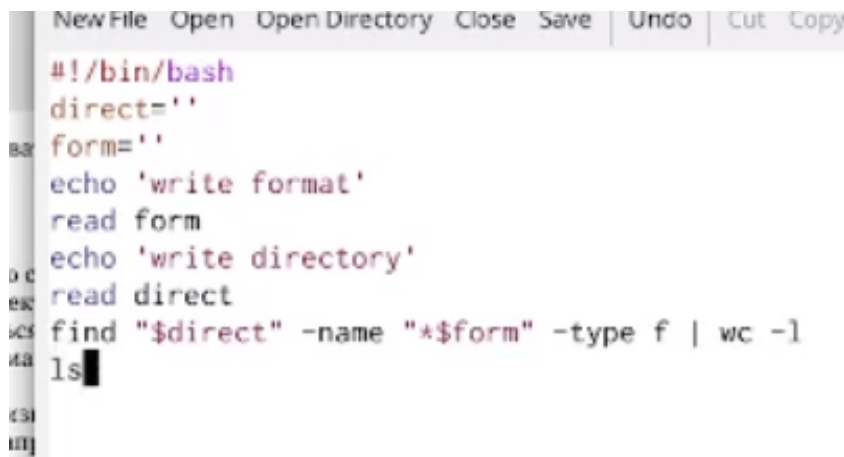
8. Проверяю его работу (рис. 3.4).



```
nykozlova@dk8n57 ~ $ ./script.sh
backup: is a directory
bin: is a directory
blog: is a directory
GNUstep: is a directory
KozlovaNonnaNBI.github.io: is a directory
public: is a directory
public_html: is a directory
script.sh: is a file andwriteable
script.sh-: is a file andwriteable
tmp: is a directory
work: is a directory
Видео: is a directory
Документы: is a directory
Загрузки: is a directory
Изображения: is a directory
Музыка: is a directory
Общедоступные: is a directory
./script.sh: строка 3: test: Рабочий: ожидается бинарный оператор
Рабочий стол: is a file and./script.sh: строка 6: test: Рабочий: ожидается бинарный оператор
./script.sh: строка 8: teat: команда не найдена
neither readable not writeable
Шаблоны: is a directory
nykozlova@dk8n57 ~ $
```

Рис. 3.4: работает

9. Пишу командный файл, который получает в качестве аргумента формат файла и вычисляет количество таких файлов в указанной директории. (рис. ??).

A screenshot of a text editor window. The menu bar at the top includes 'NewFile', 'Open', 'Open Directory', 'Close', 'Save', 'Undo', 'Cut', and 'Copy'. The script content is as follows:

```
#!/bin/bash
direct=''
form=''
echo 'write format'
read form
echo 'write directory'
read direct
find "$direct" -name "$form" -type f | wc -l
ls
```

10. Проверяю его работу (рис. ??).

A screenshot of a terminal window. The prompt is 'root@kali:~#'. The user has entered the command './script.sh'. The output of the script is displayed on the next line:

```
write format
0
write directory
4
ls
```

4 Выводы

В ходе лабораторной работы я изучила основы программирования в оболочке ОС UNIX/Linux, научилась писать небольшие командные файлы.

Список литературы