

Отчет по лабораторной работе №11

Дисциплина: Операционные системы

Чекалова Лилия Руслановна, ст.б. 1032201654

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Выводы	10
4	Библиография	11

List of Figures

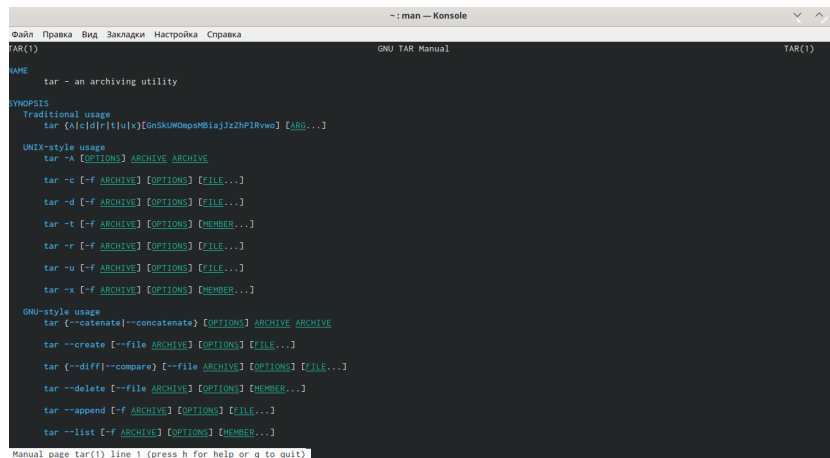
2.1	Справка по tar	5
2.2	Создание резервной копии	6
2.3	Изменение права на исполнение	6
2.4	Запуск командного файла	6
2.5	Проверка наличия архива	7
2.6	Второе задание	7
2.7	Результат работы	7
2.8	Аналог ls	8
2.9	Результат работы аналога ls	8
2.10	Подсчет файлов по формату	9
2.11	Результат поиска	9

1 Цель работы

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

2 Выполнение лабораторной работы

Читаю справку по команде tar, чтобы понять, какая у нее структура (рис. 2.1)



```
tar - an archiving utility

NAME
tar (1)

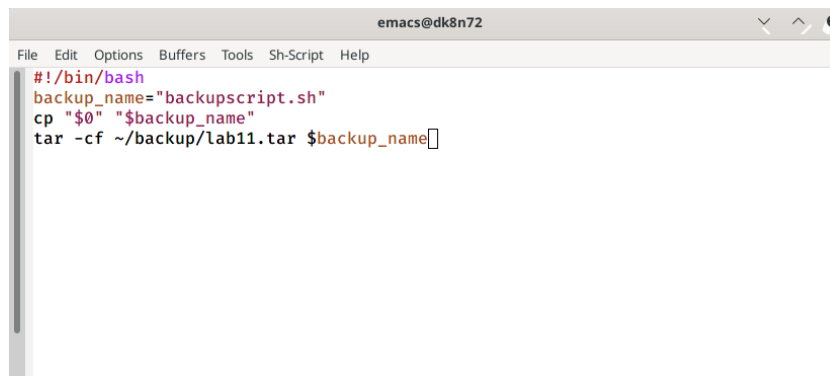
SYNOPSIS
tar (A|c|d|p|t|u|x)[G|nSkW|opsM|iajJzHpRvwo] [ABG...]

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...]
tar -r [-f ARCHIVE] [OPTIONS] [FILE...]
tar -u [-f ARCHIVE] [OPTIONS] [FILE...]
tar -x [-f ARCHIVE] [OPTIONS] [MEMBER...]

GNU-style usage
tar (--concatenate|--concatenate) [OPTIONS] ARCHIVE ARCHIVE
tar --create [--file ARCHIVE] [OPTIONS] [FILE...]
tar (--diff|--compare) [--file ARCHIVE] [OPTIONS] [FILE...]
tar --delete [--file ARCHIVE] [OPTIONS] [MEMBER...]
tar --append [-f ARCHIVE] [OPTIONS] [FILE...]
tar --list [-f ARCHIVE] [OPTIONS] [MEMBER...]
```

Figure 2.1: Справка по tar

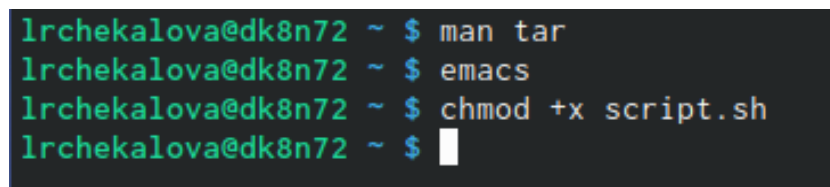
Пишу скрипт, создающий при запуске резервную копию самого себя в директорию backup, при этом файл архивируется tar. Для этого в переменную backup_name записываю название резервного файла, копирую текущий файл в резервный с помощью команды cp и передаю резервный файл команде tar для архивирования (рис. 2.2)



```
emacs@dk8n72
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
backup_name="backscript.sh"
cp "$0" "$backup_name"
tar -cf ~/backup/lab11.tar $backup_name
```

Figure 2.2: Создание резервной копии

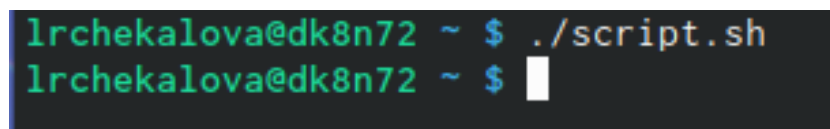
С помощью команды `chmod +x` изменяю у командного файла право на исполнение (рис. 2.3)



```
lrchekalova@dk8n72 ~ $ man tar
lrchekalova@dk8n72 ~ $ emacs
lrchekalova@dk8n72 ~ $ chmod +x script.sh
lrchekalova@dk8n72 ~ $
```

Figure 2.3: Изменение права на исполнение

Запускаю командный файл командой `./script.sh` (рис. 2.4)



```
lrchekalova@dk8n72 ~ $ ./script.sh
lrchekalova@dk8n72 ~ $
```

Figure 2.4: Запуск командного файла

В файлах проверяю успешность создания копии - в папке `backup` появился нужный нам архив `lab11.tar` (рис. 2.5)

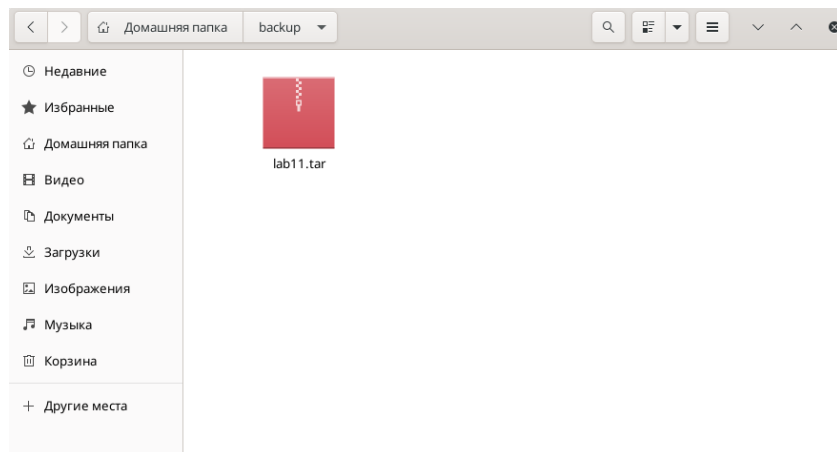


Figure 2.5: Проверка наличия архива

Пишу командный файл, обрабатывающий произвольное число аргументов, вводимое с командной строки, и распечатывающий их. Для этого вывожу подсказку для пользователя командой `echo` и использую команду `head` с опцией `-1`, чтобы указать, что на вход принимается одна строка (рис. 2.6)



Figure 2.6: Второе задание

Изменив файл на исполняемый, запускаю его и ввожу данные с клавиатуры (рис. 2.7)

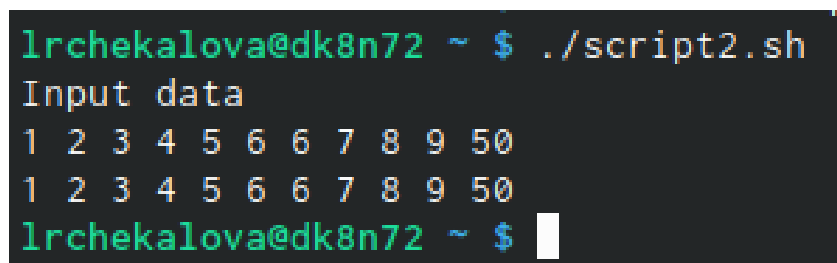
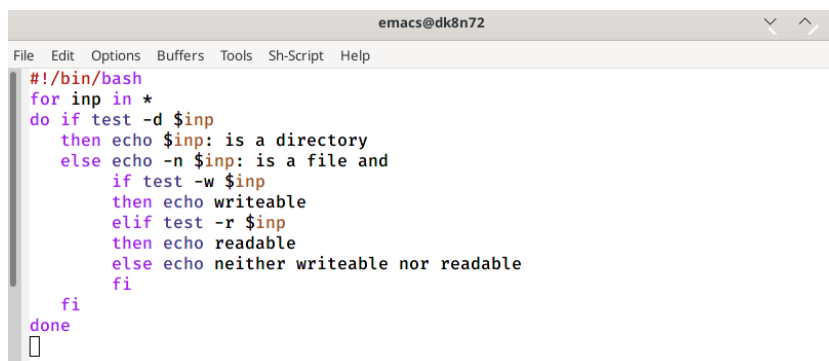


Figure 2.7: Результат работы

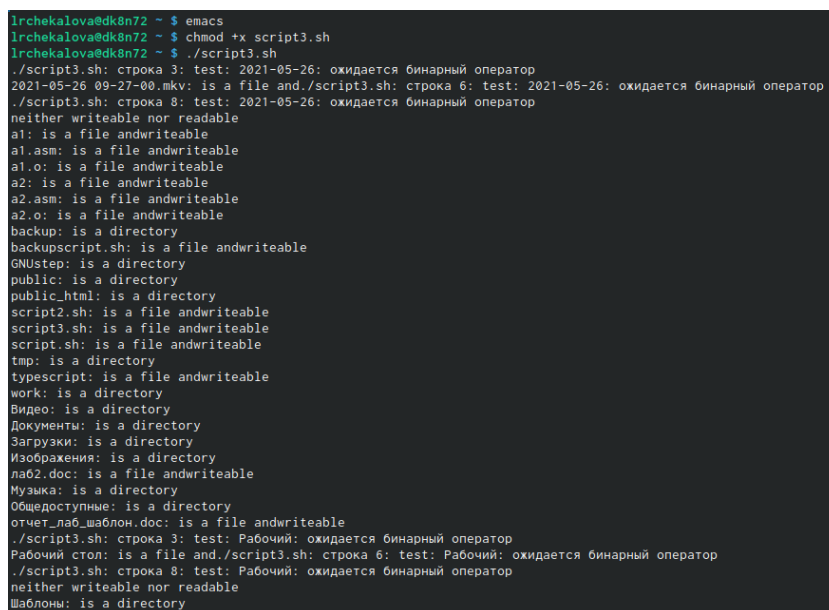
Создаю командный файл - аналог ls, сообщающий информацию о каталоге и правах доступа к файлам. Для этого использую цикл for и условный оператор if-elif-else, в котором команда test с разными опциями (-d, -w и -r) проверяет каждый поступающий объект на принадлежность к директориям или определяет права доступа к файлу, а команда echo выводит результат на экран (рис. 2.8)



```
emacs@dkn72
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
for inp in *
do if test -d $inp
then echo $inp: is a directory
else echo -n $inp: is a file and
if test -w $inp
then echo writeable
elif test -r $inp
then echo readable
else echo neither writeable nor readable
fi
fi
done
```

Figure 2.8: Аналог ls

Добавляю файлу возможность исполнения и запускаю его (рис. 2.9)



```
lrchekalova@dkn72 ~ $ emacs
lrchekalova@dkn72 ~ $ chmod +x script3.sh
lrchekalova@dkn72 ~ $ ./script3.sh
./script3.sh: строка 3: test: 2021-05-26: ожидается бинарный оператор
2021-05-26 09-27-00.mkv: is a file and./script3.sh: строка 6: test: 2021-05-26: ожидается бинарный оператор
./script3.sh: строка 8: test: 2021-05-26: ожидается бинарный оператор
neither writeable nor readable
a1: is a file andwriteable
a1.asm: is a file andwriteable
a1.o: is a file andwriteable
a2: is a file andwriteable
a2.asm: is a file andwriteable
a2.o: is a file andwriteable
backup: is a directory
backupscrip.sh: is a file andwriteable
GNUstep: is a directory
public: is a directory
public_html: is a directory
script2.sh: is a file andwriteable
script3.sh: is a file andwriteable
script.sh: is a file andwriteable
tmp: is a directory
typescript: is a file andwriteable
work: is a directory
Видео: is a directory
Документы: is a directory
Загрузки: is a directory
Изображения: is a directory
лаб2.doc: is a file andwriteable
Музыка: is a directory
Общедоступные: is a directory
отчет_лаб_шаблон.doc: is a file andwriteable
./script3.sh: строка 3: test: Рабочий: ожидается бинарный оператор
Рабочий стол: is a file and./script3.sh: строка 6: test: Рабочий: ожидается бинарный оператор
./script3.sh: строка 8: test: Рабочий: ожидается бинарный оператор
neither writeable nor readable
Шаблоны: is a directory
```

Figure 2.9: Результат работы аналога ls

Пишу командный файл, принимающий на вход формат и название директории и подсчитывающий, сколько в этой директории файлов такого формата. Для этого

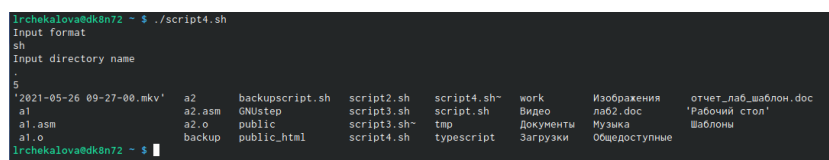
использую команду `find` с опциями `-maxdepth 1` (ограничивает поиск указанной директорией), `-name "*.format"` (критерий поиска - любое имя и определенный формат) и `-type f` (поиск по файлам). Для подсчета найденных файлов результаты работы `find` конвейером отправляю в команду `wc` с опцией `-l`, считающую количество строчек. Для наглядности вывожу содержимое указанной директории командой `ls` (рис. 2.10)



```
#!/bin/bash
format=""
directory=""
echo "Input format"
read format
echo "Input directory name"
read directory
find "$directory" -maxdepth 1 -name "*.format" -type f | wc -l
ls
```

Figure 2.10: Подсчет файлов по формату

Делаю файл исполняемым, запускаю и произвожу поиск файлов с форматом `.sh` в текущей директории (рис. 2.11)



```
lrchekalova@dk8n72 ~ $ ./script4.sh
Input format
sh
Input directory name
5
'2021-05-26 09-27-00.mkv'  a2      backups  script2.sh  script3.sh  script4.sh  work  Изображения  отчет_лаб_шаблон.doc
a1      a1.asm  a2.o     public     script3.sh  tmp         Видео  лаб2.doc     'Рабочий стол'
a1.o    backup  public_html  script4.sh  typescript  Документы  Музыка  Шаблоны
lrchekalova@dk8n72 ~ $
```

Figure 2.11: Результат поиска

3 Выводы

После выполнения данной лабораторной работы я познакомилась с переменными, циклами `for` и `while`, условным оператором `if-elif-else` и командой `test` и научилась писать небольшие командные файлы.

4 Библиография

1. Команда wc в Linux: <https://losst.ru/komanda-wc-v-linux>
2. Теоретические материалы к лабораторной работе: https://esystem.rudn.ru/pluginfile.php/1111111/mod_resource/content/1/lab_shell_prog_1.pdf
3. Циклы BASH: <https://losst.ru/tsikly-bash>