

# **отчёта по лабораторной работе №11**

**Программирование в командном процессоре ОС UNIX. Командные  
файлы**

Джумаев Бегенч

# Содержание

1	Цель работы	5
2	Задание 1	6
3	Выполнение лабораторной работы	7
4	Выводы	10

## List of Tables

# List of Figures

# 1 Цель работы

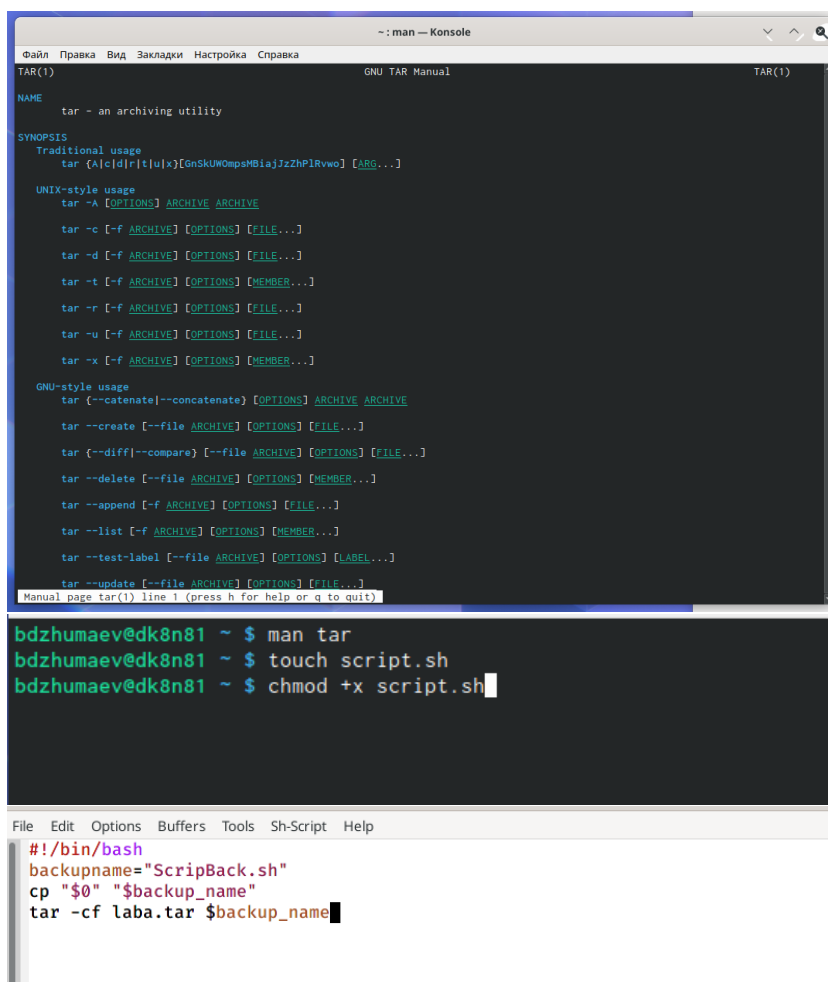
Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы

## 2 Задание 1

1. Написать скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию 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

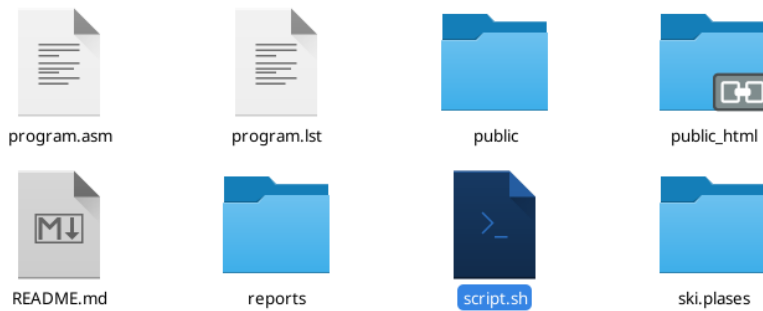
1. Я написал скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в моем каталоге. При этом файл должен архивироваться одним из архиваторов на выбор zip, bzip2 или tar. Способ использования команд архивации необходимо узнать, изучив справку.



```
~: man — Konsole
Файл  Правка  Вид  Закладки  Настройка  Справка
TAR(1)                                     GNU TAR Manual                                     TAR(1)
NAME
    tar - an archiving utility
SYNOPSIS
    Traditional usage
    tar {A|c|d|r|t|u|x}[nSkUWmpsMBiaJzZhPlRvwo] [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...]
    tar -r [-f ARCHIVE] [OPTIONS] [FILE...]
    tar -u [-f ARCHIVE] [OPTIONS] [FILE...]
    tar -x [-f ARCHIVE] [OPTIONS] [MEMBER...]
    GNU-style usage
    tar [--catenate|--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...]
    tar --test-label [--file ARCHIVE] [OPTIONS] [LABEL...]
    tar --update [--file ARCHIVE] [OPTIONS] [FILE...]
Manual page tar(1) line 1 (press h for help or q to quit)

bdzhumaev@dk8n81 ~ $ man tar
bdzhumaev@dk8n81 ~ $ touch script.sh
bdzhumaev@dk8n81 ~ $ chmod +x script.sh

File  Edit  Options  Buffers  Tools  Sh-Script  Help
#!/bin/bash
backupname="ScripBack.sh"
cp "$0" "$backup_name"
tar -cf laba.tar $backup_name
```



2. Написал пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов.

```
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
echo "Vvodite znachenie"
head -1

bdzhumaev@dk8n81 ~ $ emacs script2.sh
bdzhumaev@dk8n81 ~ $ ./script2.sh
Vvodite znachenie
1 2 3 4 5
1 2 3 4 5
```

3. Написал 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.

```
File Edit Options Buffers Tools Sh-Script Help
#!/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 test -r $A
then echo readable
else echo neither readable nor writeable
fi
fi
done
```



```

bdzhumaev@dk8n81 ~ $ ./file.sh
AA: is a directory
abc1: is a file andwriteable
asdfg: is a file andwriteable
asdfg.asm: is a file andwriteable
file.sh: is a file andwriteable
file.sh~: is a file andwriteable
GNUstep: is a directory
#lab10.sh#: is a file andwriteable
lab10.sh: is a file andwriteable
lab10.sh~: is a file andwriteable
laboratory: is a directory
main.asm: is a file andwriteable
main.lst: is a file andwriteable
Makefile: is a file andwriteable
may: is a file andwriteable
memos: is a directory
misk: is a directory
nasm.asm: is a file andwriteable
nasm.lst: is a file andwriteable
program: is a file andwriteable
program.asm: is a file andwriteable
program.lst: is a file andwriteable
public: is a directory
public_html: is a directory
README.md: is a file andwriteable
reports: is a directory
script2.sh: is a file andwriteable
script2.sh~: is a file andwriteable
script.sh: is a file andwriteable
script.sh~: is a file andwriteable
ski.plases: is a directory
tmp: is a directory
work: is a directory
Видео: is a directory
Документы: is a directory
Загрузки: is a directory

```

4. Написал командный файл, который получает в качестве аргумента командной строки формат файла (.txt,.doc,.jpg,.pdfи т.д.) и вычисляет количество таких файлов в указанной директории. Путь к директории также передаётся в виде аргумента командной строки

```

File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
format=""
direct=""
echo "point out format"
read format
echo "point out direct"
read direct
find "$direct" -name ".$format" -type f | wc -l
ls

```

```

bdzhumaev@dk8n81 ~ $ touch script3.sh
bdzhumaev@dk8n81 ~ $ chmod +x script3.sh
bdzhumaev@dk8n81 ~ $ emacs script3.sh
bdzhumaev@dk8n81 ~ $ emacs script3.sh
bdzhumaev@dk8n81 ~ $ ./script3.sh
point out format
.sh
point out direct
/home/bdzhumaev/
find: '/home/bdzhumaev/': Нет такого файла или каталога

```

AA	file.sh~	laboratory	memos	program.asm	reports	script.sh	Видео	Общедоступные
abc1	GNUstep	main.asm	misk	program.lst	script2.sh	script.sh~	Документы	'Рабочий стол'
asdfg	#lab10.sh#	main.lst	nasm.asm	public	script2.sh~	ski.plases	Загрузки	Шаблоны
asdfg.asm	lab10.sh	Makefile	nasm.lst	public_html	script3.sh	tmp	Изображения	
file.sh	lab10.sh~	may	program	README.md	script3.sh~	work	Музыка	

## 4 Выводы

Я изучал основы программирования в оболочке ОС UNIX/Linux и научился писать небольшие командные файлы.