

Лабораторная работа №10

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

Зарифбеков Амир Пайшанбиевич

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	11

Список иллюстраций

3.1	написал скрипт	7
3.2	как он работает	8
3.3	написали скрипт	8
3.4	то как он работает	8
3.5	написсал файл	9
3.6	смотрим как он работает	9
3.7	написал файл	10
3.8	как он работет	10

Список таблиц

1 Цель работы

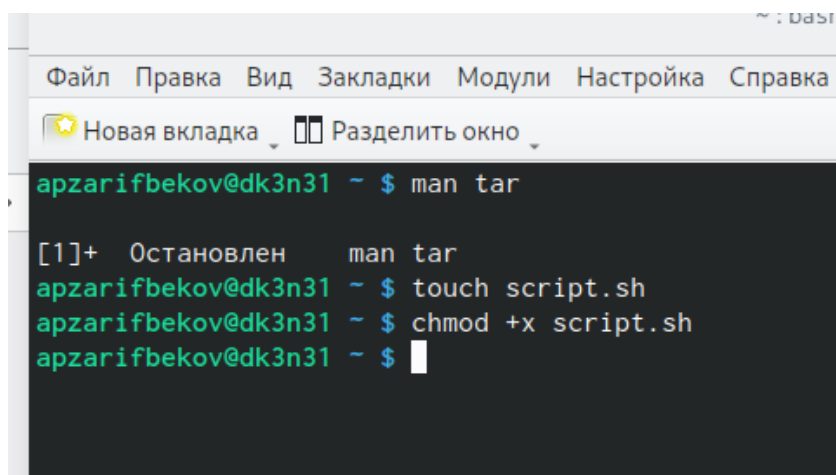
Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.

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

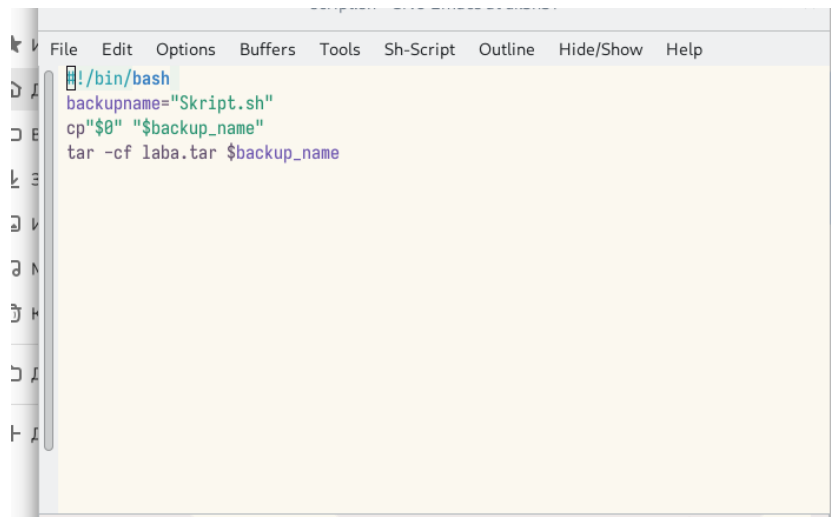
1. Написал скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в нашем домашнем каталоге.



The image shows a terminal window with a menu bar at the top containing 'Файл', 'Правка', 'Вид', 'Закладки', 'Модули', 'Настройка', and 'Справка'. Below the menu bar, there are two buttons: 'Новая вкладка' (with a star icon) and 'Разделить окно' (with a window icon). The terminal text shows the user 'apzarifbekov@dk3n31' running the following commands:

```
apzarifbekov@dk3n31 ~ $ man tar  
[1]+  Остановлен      man tar  
apzarifbekov@dk3n31 ~ $ touch script.sh  
apzarifbekov@dk3n31 ~ $ chmod +x script.sh  
apzarifbekov@dk3n31 ~ $
```

Рис. 3.1: написал скрипт



```
#!/bin/bash
backupname="Skript.sh"
cp "$@" "$backup_name"
tar -cf laba.tar $backup_name
```

Рис. 3.2: как он работает

2. Написал пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять



```
#!/bin/bash
echo "Vvedite znachenie"
head -1
```

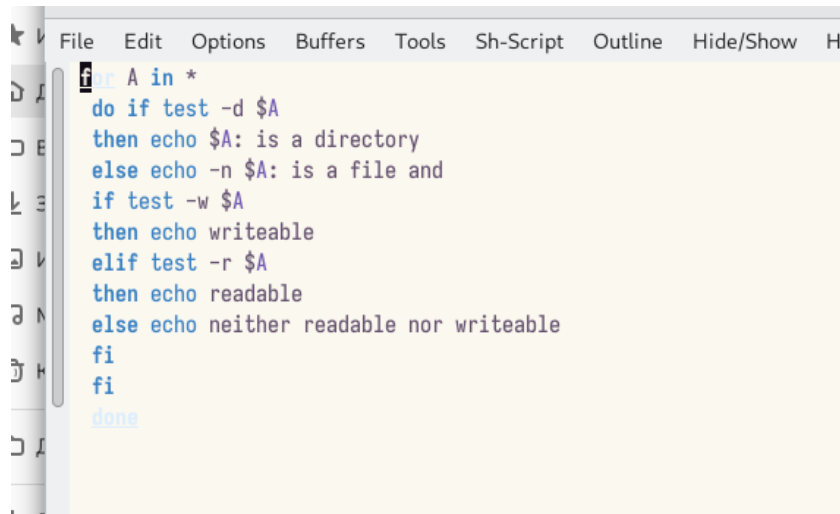
- 45 script2.sh Shell-script unix | 1: 0 All
Welcome to GNU Emacs, one component of the GNU/Linux operating system.

Рис. 3.3: написали скрипт

то как он работает

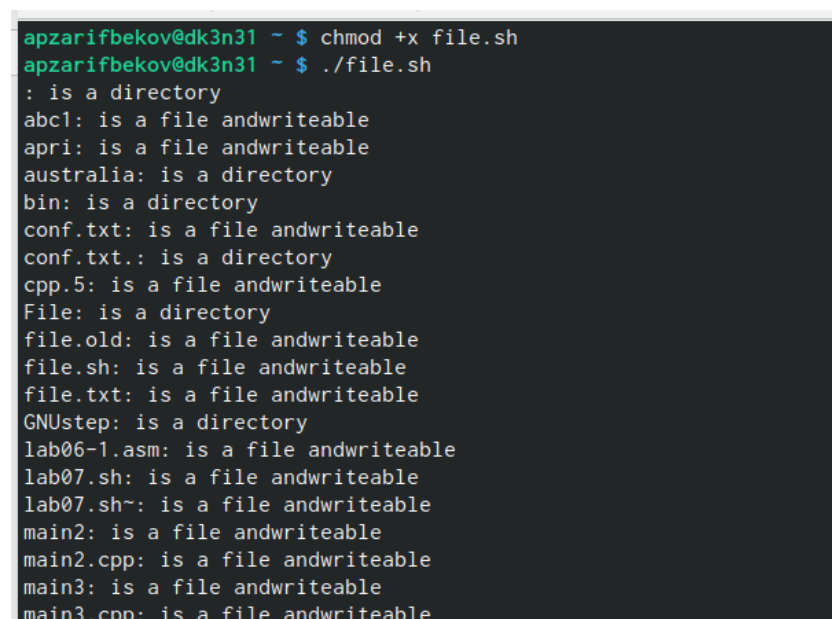
Рис. 3.4: то как он работает

3. Написал командный файл — аналог команды ls

A screenshot of a text editor window with a menu bar (File, Edit, Options, Buffers, Tools, Sh-Script, Outline, Hide/Show, Help) and a sidebar. The editor contains a shell script for listing files and their permissions. The script uses a 'for' loop to iterate over files in the current directory, and nested 'if' statements to check if a file is a directory, and if it's a file, whether it's readable or writeable. The output is formatted to show the file name followed by its type and permissions.

```
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
```

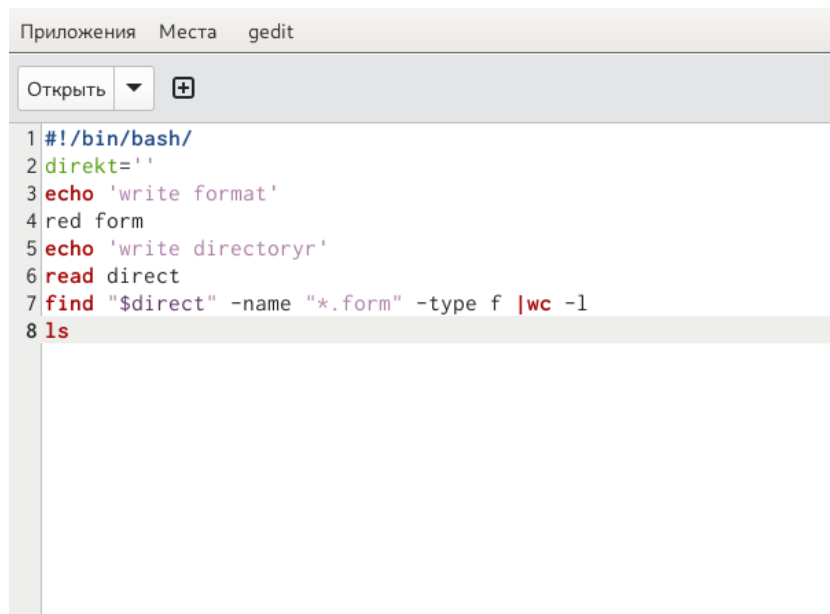
Рис. 3.5: написал файл

A screenshot of a terminal window showing the execution of the 'file.sh' script. The user first makes the script executable with 'chmod +x file.sh' and then runs it with './file.sh'. The output lists various files and directories in the current directory, indicating whether they are directories or files, and if files, whether they are readable or writeable.

```
apzarifbekov@dk3n31 ~ $ chmod +x file.sh
apzarifbekov@dk3n31 ~ $ ./file.sh
: is a directory
abc1: is a file andwriteable
apri: is a file andwriteable
australia: is a directory
bin: is a directory
conf.txt: is a file andwriteable
conf.txt.: is a directory
cpp.5: is a file andwriteable
File: is a directory
file.old: is a file andwriteable
file.sh: is a file andwriteable
file.txt: is a file andwriteable
GNUstep: is a directory
lab06-1.asm: is a file andwriteable
lab07.sh: is a file andwriteable
lab07.sh~: is a file andwriteable
main2: is a file andwriteable
main2.cpp: is a file andwriteable
main3: is a file andwriteable
main3.cpp: is a file andwriteable
```

Рис. 3.6: смотрим как он работает

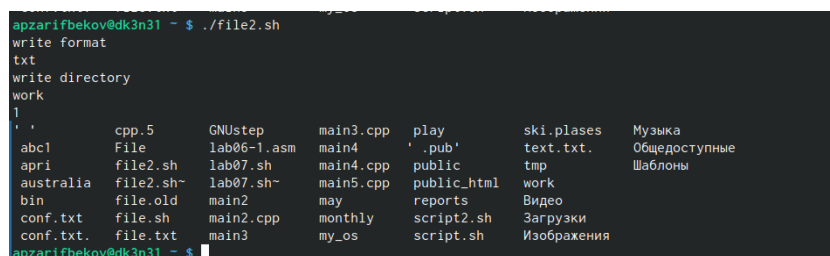
4. Написал командный файл, который получает в качестве аргумента командной строки формат файла и вычисляет количество таких файлов в указанной директории.



The screenshot shows a gedit text editor window with a menu bar containing 'Приложения', 'Места', and 'gedit'. Below the menu bar is a toolbar with an 'Открыть' button and a '+' icon. The main text area contains a shell script with the following lines:

```
1 #!/bin/bash/  
2 direkt=' '  
3 echo 'write format'  
4 red form  
5 echo 'write directoryr'  
6 read direct  
7 find "$direct" -name "*.form" -type f |wc -l  
8 ls
```

Рис. 3.7: написал файл



The screenshot shows a terminal window with the prompt 'apzarifbekov@dk3n31 ~ \$'. The user has executed the command './file2.sh'. The output of the script is as follows:

```
apzarifbekov@dk3n31 ~ $ ./file2.sh  
write format  
txt  
write directory  
work  
1  
' '  
abcl      cpp.5      GNUstep    main3.cpp  play       ski.plases  Музыка  
File      lab06-1.asm  main4     '.pub'    text.txt.  Общедоступные  
apri      file2.sh     lab07.sh  main4.cpp  public     tmp         Шаблоны  
australia file2.sh~   lab07.sh~ main5.cpp  public_html work  
bin       file.old    main2     may        reports    Видео  
conf.txt  file.sh     main2.cpp monthly   script2.sh Загрузки  
conf.txt. file.txt    main3     my_os     script.sh  Изображения  
apzarifbekov@dk3n31 ~ $
```

Рис. 3.8: как он работает

4 Выводы

Я изучил основы программирования в оболочке ОС UNIX/Linux. Научился писать небольшие командные файлы