

# Лабораторная работа №10

Программирование в командном процессоре ОС UNIX. Командные файлы

---

Зарифбеков А. П.

13 апреля 2023

Российский университет дружбы народов, Москва, Россия

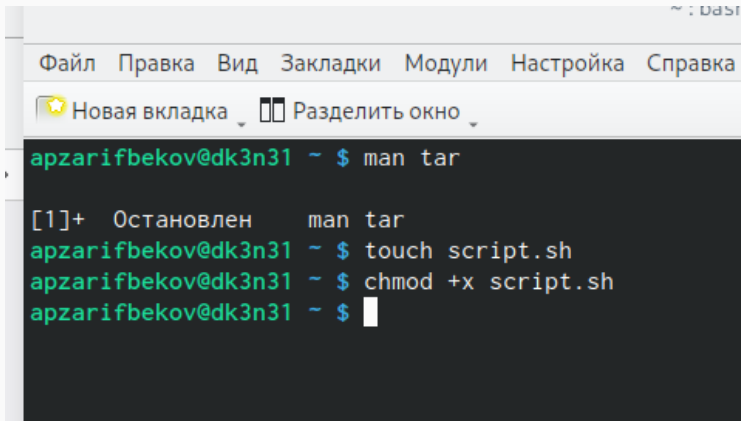
- Зарифбеков Амир Пайшанбиевич
- Студент НБИбд-01-22
- Российский университет дружбы народов

- Актуально для тех кто хочет изучить программирование в командном процессоре ОС UNIX и командные файлы

- Презентация как текст
- Программное обеспечение для создания презентаций
- Входные и выходные форматы презентаций

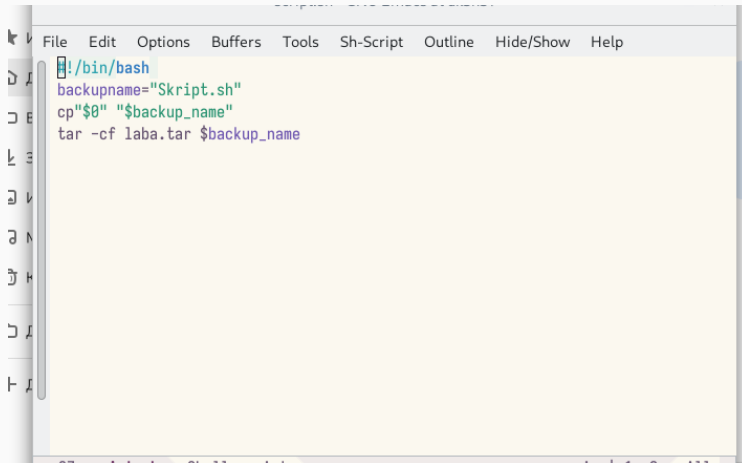
- Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.
- Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.
- Написать скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, 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.
- 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.
- Написать командный файл — аналог команды ls (без использования самой этой команды и команды dir). Требуется, чтобы он выдавал информацию о нужном каталоге и

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 screenshot shows a terminal window with a menu bar at the top containing 'Файл', 'Правка', 'Вид', 'Закладки', 'Модули', 'Настройка', and 'Справка'. Below the menu bar is a toolbar with a star icon and the text 'Новая вкладка', followed by a window icon and 'Разделить окно'. The terminal content shows the user 'apzarifbekov@dk3n31' in the home directory '~' running the command 'man tar'. The output is '[1]+ Остановлен man tar'. The user then runs 'touch script.sh', 'chmod +x script.sh', and finally '\$' with a cursor, indicating the prompt is ready for the next command.

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

A screenshot of a terminal window with a menu bar at the top containing 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', 'Outline', 'Hide/Show', and 'Help'. The terminal has a yellow background and shows a blue prompt character followed by the command `#!/bin/bash`. Below this, three lines of script are displayed: `backupname="Skript.sh"`, `cp "$@" "$backup_name"`, and `tar -cf laba.tar $backup_name`. A vertical scrollbar is visible on the left side of the terminal window.

```
File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help
#!/bin/bash
backupname="Skript.sh"
cp "$@" "$backup_name"
tar -cf laba.tar $backup_name
```

Рис. 2: как он работает

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



```
File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help
#!/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.
```

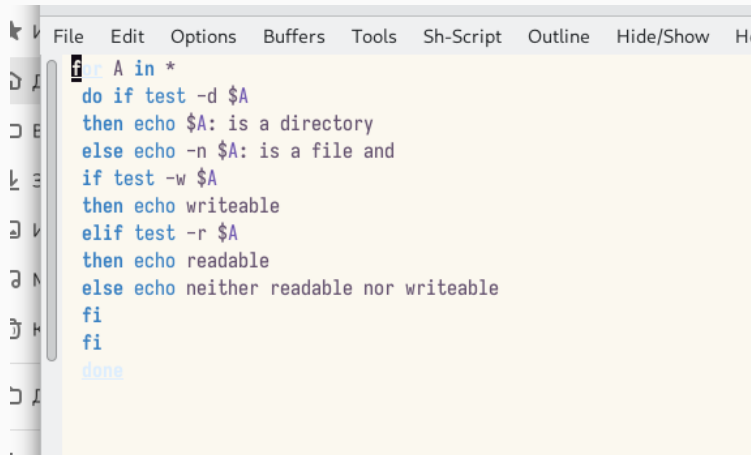
Рис. 3: написали скрипт



то как он работает

**Рис. 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 toolbar on the left. The editor contains a shell script for the 'ls' command. The script uses a 'for' loop to iterate over files, and an 'if' statement to check if a file is a directory. If it is a directory, it prints the file name followed by 'is a directory'. If it is a file, it checks if the file is writeable or readable and prints the appropriate message. The script ends with 'done'.

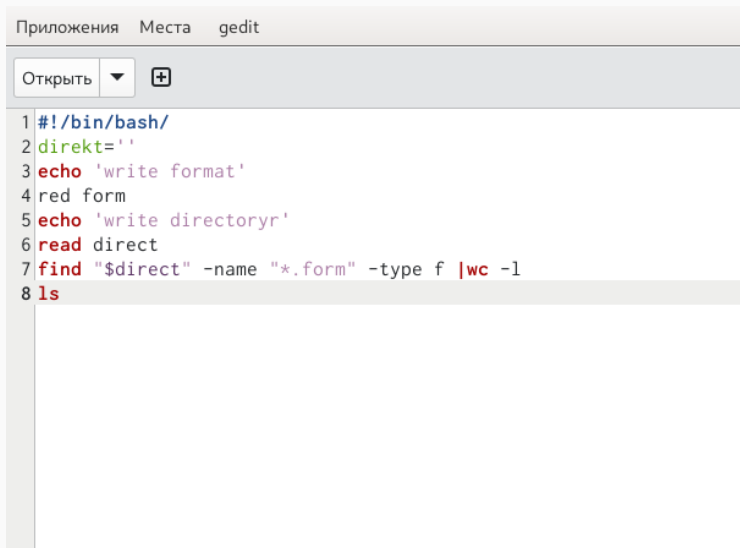
```
for A in *
do if test -d $A
then echo $A: is a directory
else echo -n $A: is a file and
if test -w $A
then echo writeable
elif test -r $A
then echo readable
else echo neither readable nor writeable
fi
fi
done
```

Рис. 5: написал файл

```
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. 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 image shows a screenshot of a text editor window titled "gedit". The window has a menu bar with "Приложения", "Места", and "gedit". Below the menu bar is a toolbar with a button labeled "Открыть" (Open), a dropdown arrow, and a plus icon. The main area of the window contains a shell script with 8 lines of code, each preceded by a line number in a light gray margin. The code is as follows:

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

```

apzarifbekov@dk3n31 ~ $ ./file2.sh
write format
txt
write directory
work
1
' '
      cpp.5      GNUstep      main3.cpp      play      ski.plases      Музыка
abc1      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. 8: как он работает

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