

Лабораторная работа 12

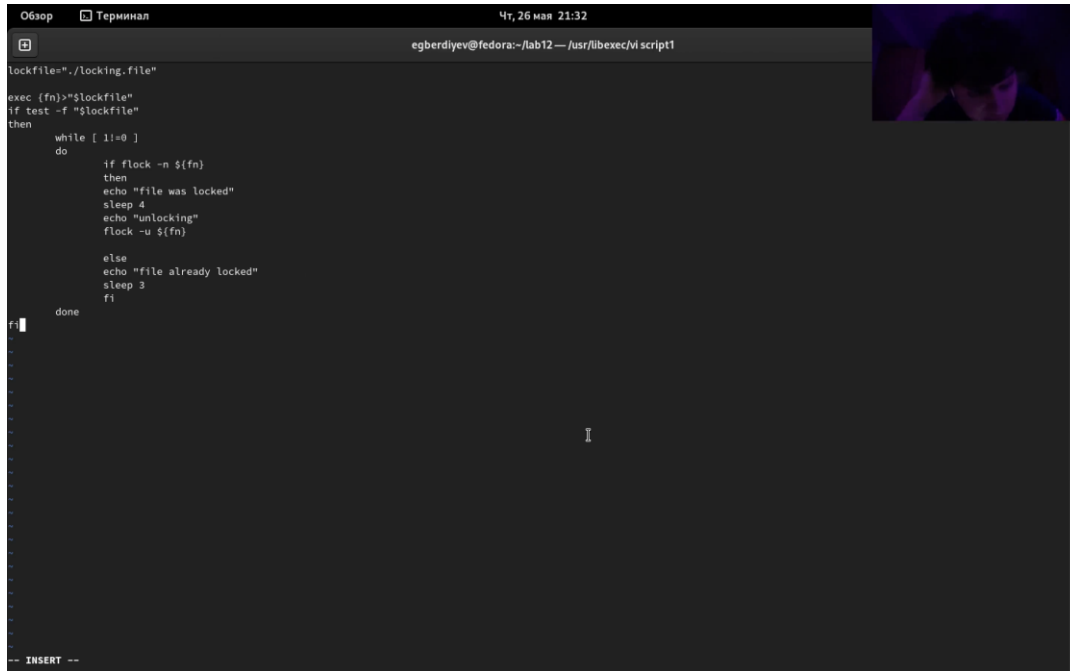
Бердыев Эзиз Группа НФИбд-01-21

Цель работы

Изучить основы программирования в оболочке ОС UNIX.
Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Выполнение

1. Скрипт 1



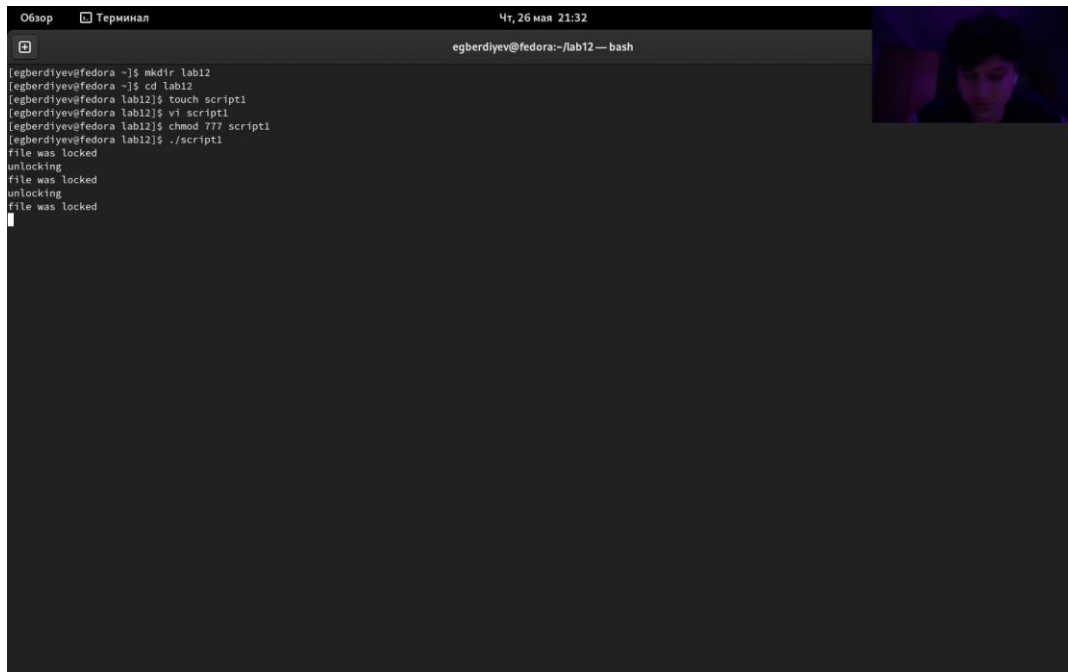
```
lockfile="/locking.file"

exec {fn}>"$lockfile"
if test -f "$lockfile"
then
    while [ 1!=0 ]
    do
        if flock -n ${fn}
        then
            echo "file was locked"
            sleep 4
            echo "unlocking"
            flock -u ${fn}

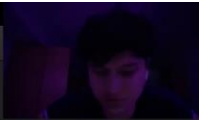
        else
            echo "file already locked"
            sleep 3
        fi
    done
fi
```

Скрипт 1.png

Скрипт 1 запуск

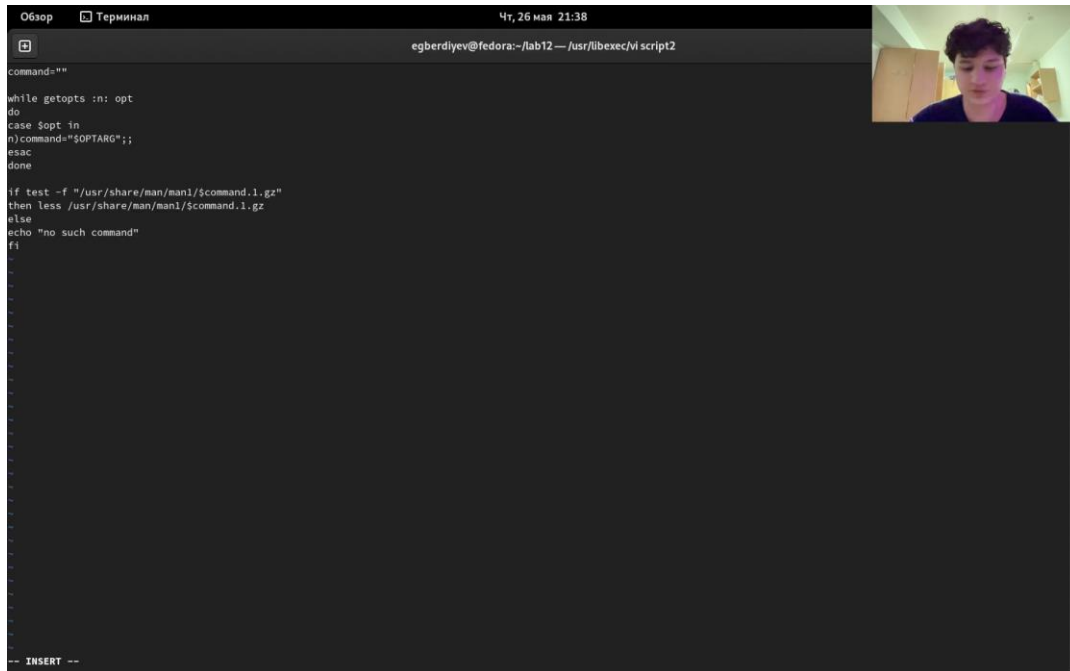
A terminal window titled "Обзор Терминал" with a timestamp of "Чт, 26 мая 21:32". The prompt is "egberdiyev@fedora:~/lab12 — bash". The terminal shows the following commands and output:

```
[egberdiyev@fedora ~]$ mkdir lab12
[egberdiyev@fedora ~]$ cd lab12
[egberdiyev@fedora lab12]$ touch script1
[egberdiyev@fedora lab12]$ vi script1
[egberdiyev@fedora lab12]$ chmod 777 script1
[egberdiyev@fedora lab12]$ ./script1
file was locked
unlocking
file was locked
unlocking
file was locked
```

A small video feed in the top right corner of the terminal window shows a person with dark hair looking down at a screen.

Скрипт 1 запуск.png

2. Скрипт 2



```
Обзор Терминал 4т, 26 мая 21:38
egberdiyev@fedora:~/lab12 — /usr/libexec/ls script2

command=""

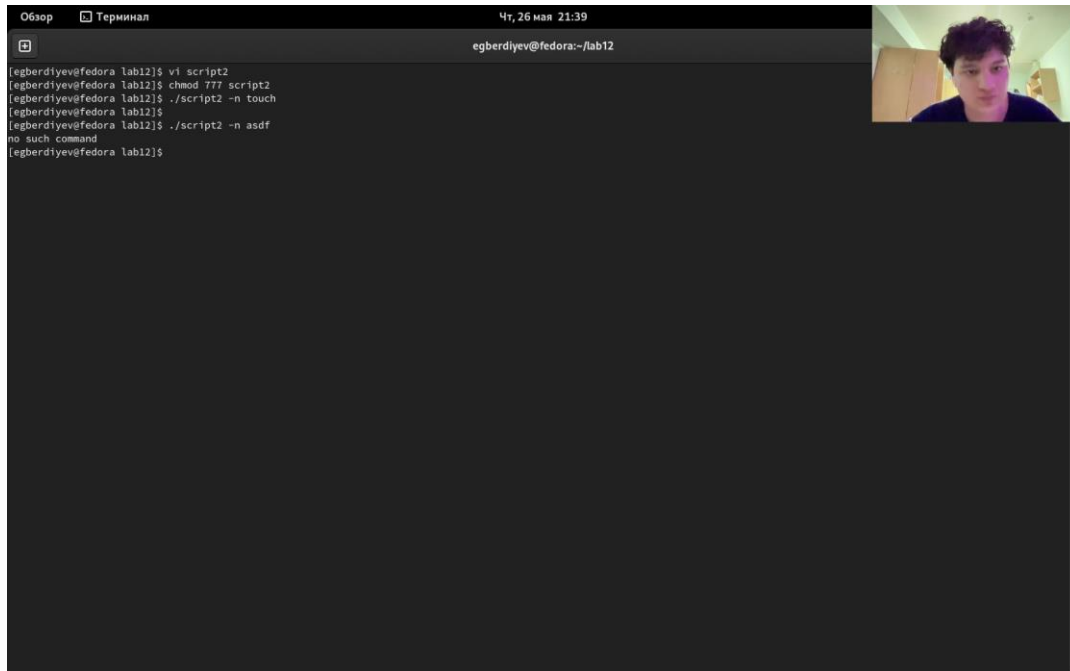
while getopts :n: opt
do
case $opt in
n)command="$OPTARG";;
esac
done

if test -f "/usr/share/man/man1/$command.1.gz"
then less /usr/share/man/man1/$command.1.gz
else
echo "no such command"
fi

-- INSERT --
```

Скрипт 2.png

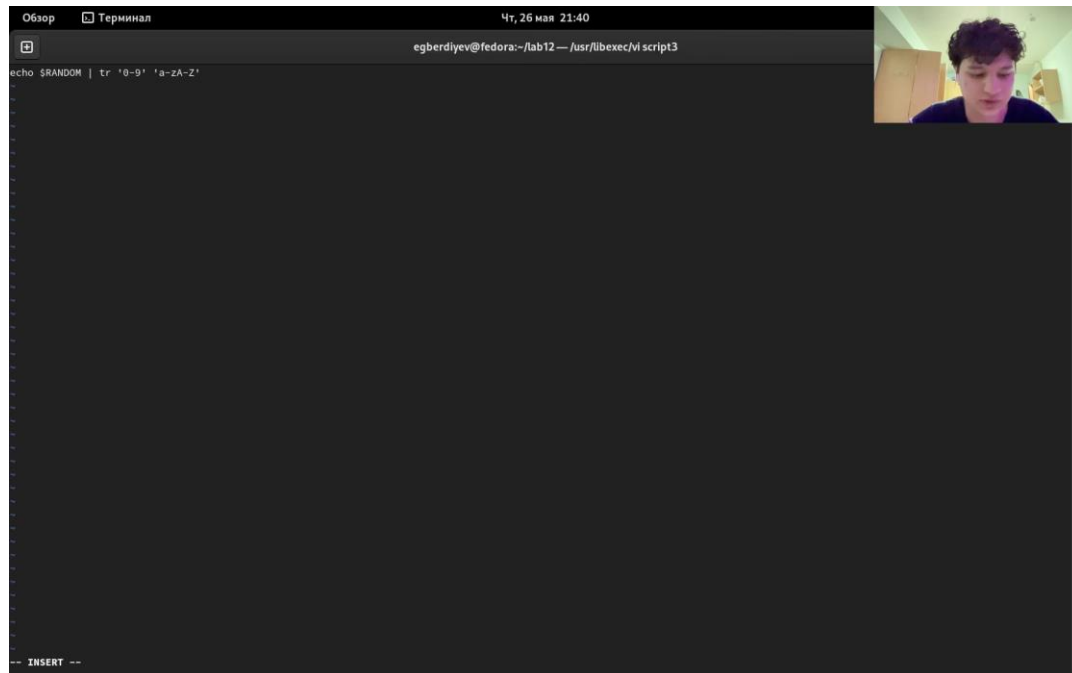
Скрипт 2 запуск



```
Обзор Терминал Чт, 26 мая 21:39 egberdiyev@fedora: ~/lab12
[egberdiyev@fedora lab12]$ vi script2
[egberdiyev@fedora lab12]$ chmod 777 script2
[egberdiyev@fedora lab12]$ ./script2 -n touch
[egberdiyev@fedora lab12]$
[egberdiyev@fedora lab12]$ ./script2 -n asdf
no such command
[egberdiyev@fedora lab12]$
```

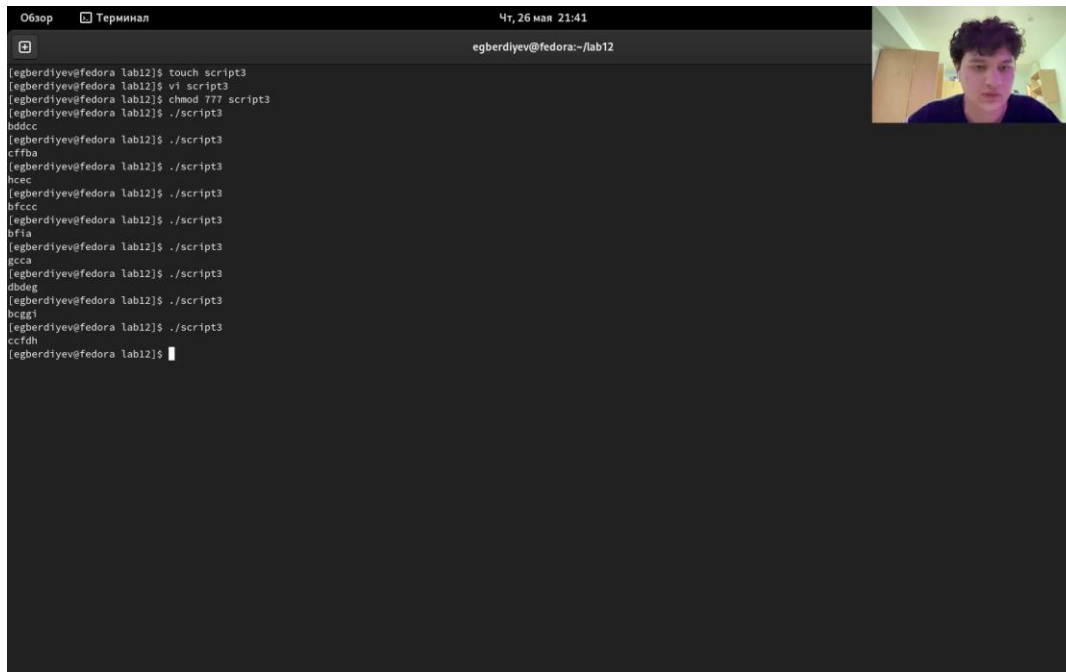
Скрипт 2 запуск.png

3. Скрипт 3



Скрипт 3.png

Скрипт 3 запуск



The image shows a terminal window with a dark background and light text. The window title bar includes 'Обзор' (Overview) and 'Терминал' (Terminal) tabs, and a timestamp 'Чт, 26 мая 21:41'. The terminal content shows a series of commands and their outputs in a Fedora environment. The commands include creating a file, setting permissions, and running a script multiple times, which produces hexadecimal-like output. In the top right corner, there is a small video feed window showing a person with curly hair.

```
egberdiyev@fedora lab12$ touch script3
egberdiyev@fedora lab12$ vi script3
egberdiyev@fedora lab12$ chmod 777 script3
egberdiyev@fedora lab12$ ./script3
b0dccc
egberdiyev@fedora lab12$ ./script3
cfffba
egberdiyev@fedora lab12$ ./script3
hccc
egberdiyev@fedora lab12$ ./script3
bfccc
egberdiyev@fedora lab12$ ./script3
bfia
egberdiyev@fedora lab12$ ./script3
gcca
egberdiyev@fedora lab12$ ./script3
dbdeg
egberdiyev@fedora lab12$ ./script3
bcgg1
egberdiyev@fedora lab12$ ./script3
ccfdh
egberdiyev@fedora lab12$
```

Скрипт 3 запуск.png

Контрольные вопросы

1. Найдите синтаксическую ошибку в следующей строке:
`while [$1 != "exit"]` Так же между скобками должны быть пробелы. В противном случае скобки и рядом стоящие символы будут восприниматься как одно целое

2. Как объединить (конкатенация) несколько строк в одну?

```
cat file.txt | xargs | sed -e 's/\n /\n/g'
```

3. Найдите информацию об утилите seq. Какими иными способами можно реализовать её функционал при программировании на bash? seq - выдает последовательность чисел. Реализовать ее функционал можно командой `for n in {1..5} do done`

4. Какой результат даст вычисление выражения $\$(10/3)$?
3

5. Укажите кратко основные отличия командной оболочки zsh от bash.

Zsh очень сильно упрощает работу. Но существуют различия. Например, в zsh после for обязательно вставлять пробел, нумерация массивов в zsh начинается с 1 (что не особо удобно на самом деле). Если вы собираетесь писать скрипт, который легко будет запускать множество разработчиков, то я рекомендую Bash. Если скрипты вам не нужны - Zsh (более простая работа с файлами, например)

6. Проверьте, верен ли синтаксис данной конструкции for
((a=1; a <= LIMIT; a++)) Верен 9

7. Сравните язык `bash` с какими-либо языками программирования. Какие преимущества у `bash` по сравнению с ними? Какие недостатки? `Bash` позволяет очень легко работать с файловой системой без лишних конструкций (в отличие от обычного языка программирования). Но относительно обычных языков программирования `bash` очень сжат. Тот же `C` имеет гораздо более широкие возможности для разработчика.

Вывод

Изучил основы программирования в оболочке ОС UNIX.
Научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.