

# Лабораторная работа №12

---

Коняева Марина НФИбд-01-21

17.05.2022

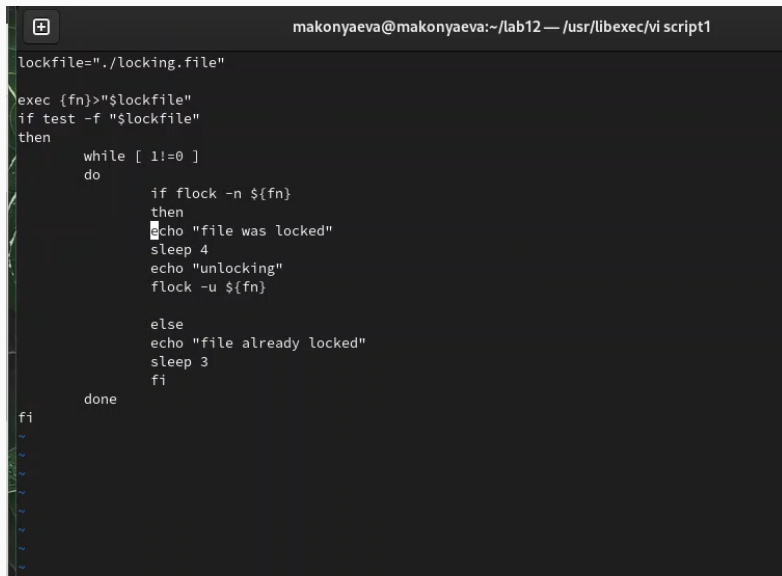
## Программирование в командном процессоре ОС UNIX. Расширенное программирование

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

Командный процессор (командная оболочка, интерпретатор команд shell) — это программа, позволяющая пользователю взаимодействовать с операционной системой компьютера. В операционных системах типа UNIX/Linux наиболее часто используются следующие реализации командных оболочек: — оболочка Борна (Bourne shell или sh) — стандартная командная оболочка UNIX/Linux, содержащая базовый, но при этом полный набор функций; — C-оболочка (или csh) — надстройка на оболочкой Борна, использующая C-подобный синтаксис команд с возможностью сохранения истории выполнения команд; — оболочка Корна (или ksh) — напоминает оболочку C, но операторы управления программой совместимы с операторами оболочки Борна; — BASH — сокращение от Bourne Again Shell (опять оболочка Борна), в основе своей совмещает свойства оболочек C и

# Выполнение лабораторной работы

## 1. Скрипт 1 (изображение 1.1-2)



```
makonyaeva@makonyaeva:~/lab12 — /usr/libexec/vi script1
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
```



## 2. Скрипт 2 (изображение 2.1-2)

```
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

~
~
~
~
~
~
~
~
~
~
~
```

```

ESC[1mNAMEESC[0m
touch - change file timestamps

ESC[1mSYNOPSISESC[0m
ESC[1mtouch ESC[22m[ESC[4mOPTIONESC[24m]... ESC[4mFILEESC[24m...

ESC[1mDESCRIPTIONESC[0m
Update the access and modification times of each FILE to the current time.

A FILE argument that does not exist is created empty, unless ESC[1m-c ESC[22mor ESC[1m-h ESC[22m

A FILE argument string of - is handled specially and causes touch to change the times of the fi
ciated with standard output.

Mandatory arguments to long options are mandatory for short options too.

ESC[1m-a ESC[22mchange only the access time

ESC[1m-dESC[22m, ESC[1m--no-createESC[0m
do not create any files

ESC[1m-dESC[22m, ESC[1m--dateESC[22m=ESC[4mSTRINGESC[0m
parse STRING and use it instead of current time

ESC[1m-f ESC[22m(ignored)

ESC[1m-hESC[22m, ESC[1m--no-dereferenceESC[0m
affect each symbolic link instead of any referenced file (useful only on systems that can
the timestamps of a symlink)

ESC[1m-m ESC[22mchange only the modification time

ESC[1m-rESC[22m, ESC[1m--referenceESC[22m=ESC[4mFILEESC[0m
use this file's times instead of current time

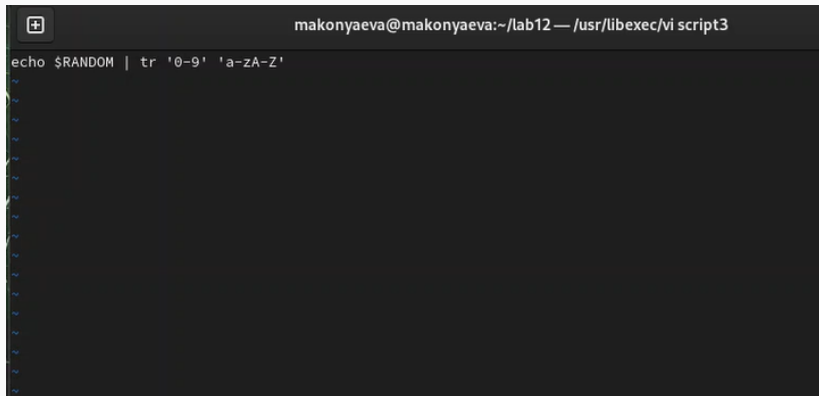
ESC[1m-t ESC[22mSTAMP
use [[CC]YY]MMDDhhmm[.ss] instead of current time

ESC[1m--timeESC[22m=ESC[4mWORDESC[0m
change the specified time: WORD is access, atime, or use: equ
modify or

```



### 3. Скрипт 3 (изображение 3.1-2)



```
makonyaeva@makonyaeva:~/lab12 — /usr/libexec/vi script3
echo $RANDOM | tr '0-9' 'a-zA-Z'
```

Изображение 3.1 Скрипт 3

```
[makonyaeva@makonyaeva lab12]$ touch script3
[makonyaeva@makonyaeva lab12]$ vi script3
[makonyaeva@makonyaeva lab12]$ chmod 777 script3
[makonyaeva@makonyaeva lab12]$ ./script3
daieb
[makonyaeva@makonyaeva lab12]$ ./script3
bjhb
[makonyaeva@makonyaeva lab12]$ ./script3
cabha
[makonyaeva@makonyaeva lab12]$ ./script3
behij
[makonyaeva@makonyaeva lab12]$
```

В ходе данной лабораторной работы изучили основы программирования в оболочке ОС UNIX. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов, а также ответили на контрольные вопросы.