

# РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

## Факультет физико-математических и естественных наук

### Кафедра прикладной информатики и теории вероятностей

#### ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 13 «Программирование в командном процессоре ОС UNIX. Расширенное программирование»

дисциплина: *Операционные системы*

Студент: Саргсян Арам Грачьевич

Группа: НПИбд 02-20

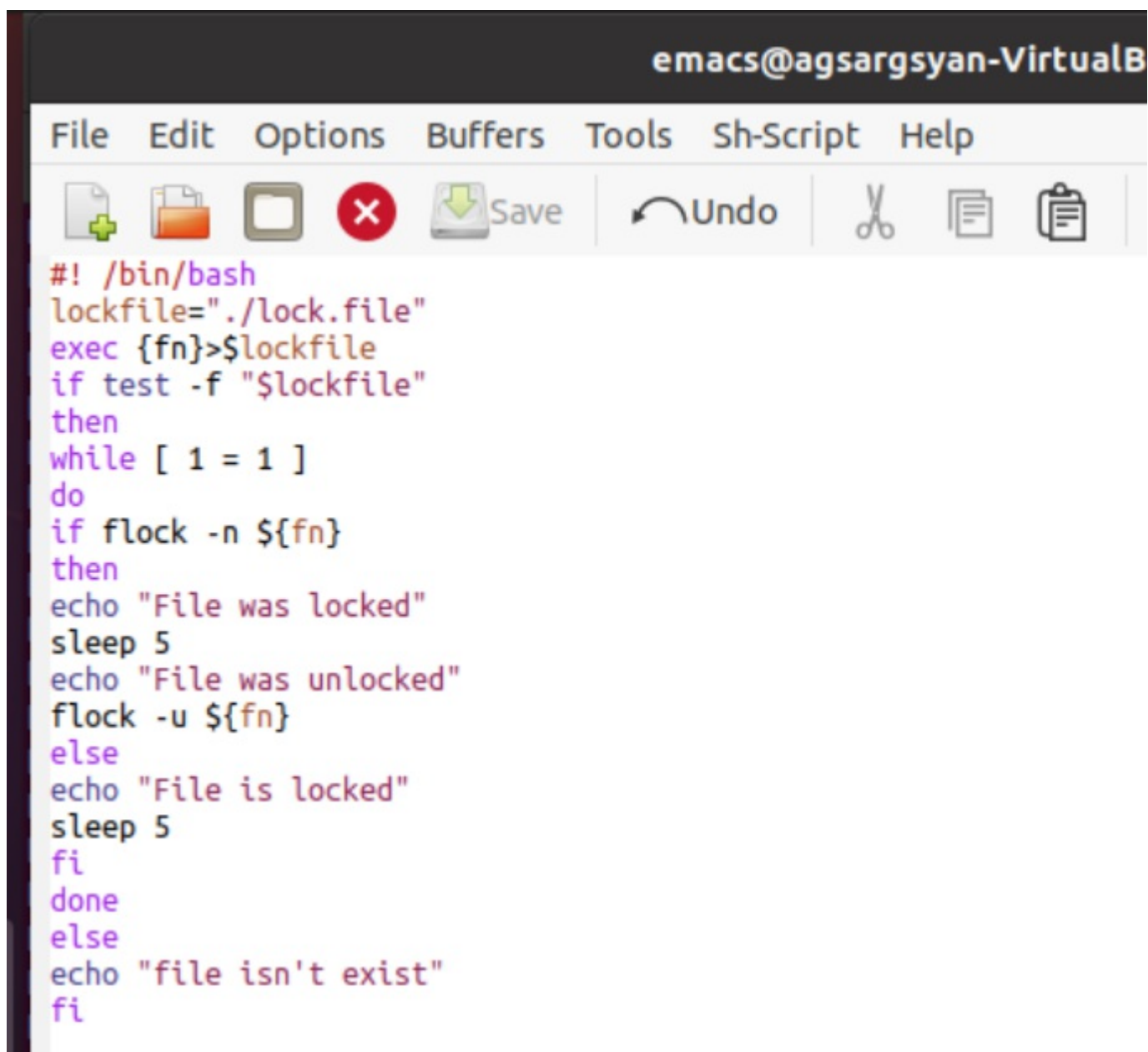
МОСКВА 2021 г.

---

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

#### Ход работы

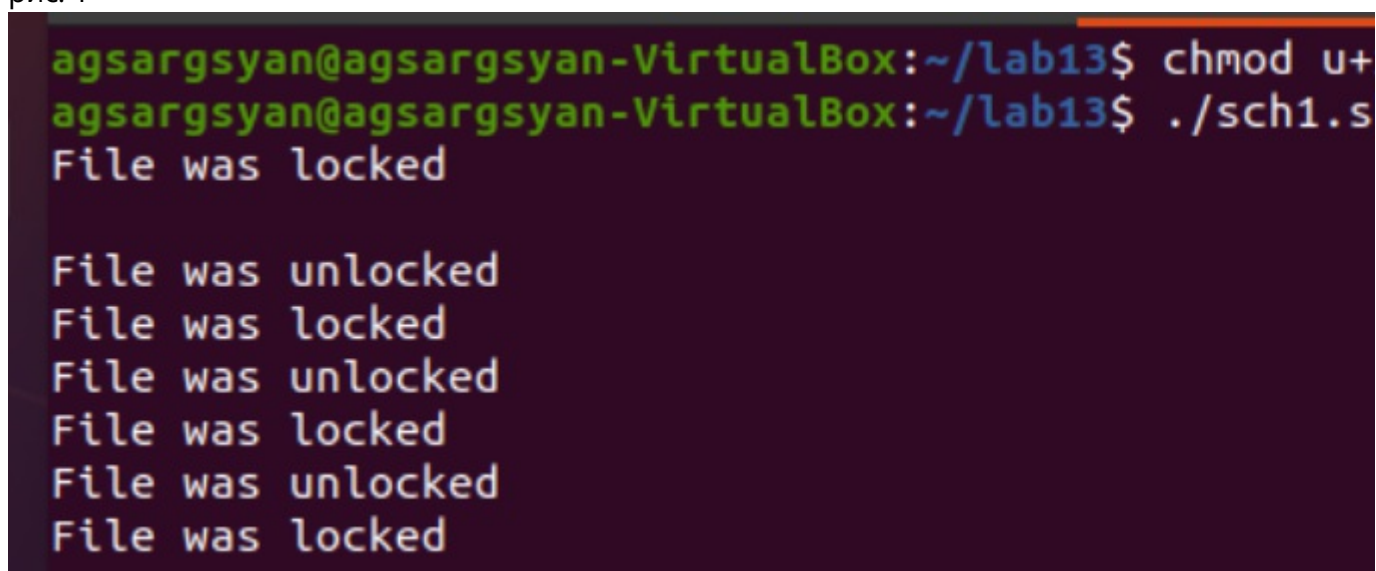
1. Написали командный файл, реализующий упрощённый механизм семафоров. Командный файл должен в течение некоторого времени  $t_1$  дожидаться освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использовать его в течение некоторого времени  $t_2 < t_1$ , также выдавая информацию о том, что ресурси спользуется соответствующим командным файлом(процессом). Запустили командный файл в одном виртуальном терминале в фоновом режиме, перенаправив его вывод в другой ( $> /dev/tty\#$ , где  $\#$  — номер терминала куда перенаправляется вывод), в котором также запущен этот файл, но не фоновом, а в привилегированном режиме. Доработали программу так, чтобы имелась возможность взаимодействия трёх и более процессов. (Рис. 1-2)



The screenshot shows the Emacs editor interface with the title bar 'emacs@agsargsyan-VirtualB'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The toolbar contains icons for file operations and editing. The main text area displays a shell script for file locking using flock.

```
#!/bin/bash
lockfile="./lock.file"
exec {fn}>$lockfile
if test -f "$lockfile"
then
while [ 1 = 1 ]
do
if flock -n ${fn}
then
echo "File was locked"
sleep 5
echo "File was unlocked"
flock -u ${fn}
else
echo "File is locked"
sleep 5
fi
done
else
echo "file isn't exist"
fi
```

рис. 1



The screenshot shows a terminal window with the prompt 'agsargsyan@agsargsyan-VirtualBox:~/lab13\$'. The user enters 'chmod u+' and then './sch1.s'. The output shows the script running, alternating between 'File was locked' and 'File was unlocked' messages.

```
agsargsyan@agsargsyan-VirtualBox:~/lab13$ chmod u+
agsargsyan@agsargsyan-VirtualBox:~/lab13$ ./sch1.s
File was locked

File was unlocked
File was locked
File was unlocked
File was locked
File was unlocked
File was locked
```

рис. 2

2. Реализуем команду тап с помощью командного файла. (рис. 3-4)

```
emacs@agsargsyan-VirtualBo
File Edit Options Buffers Tools Sh-Script Help
Save Undo
#!/bin/bash
name=""

while getopts "n:" opt
do
case $opt in
n)name="$OPTARG";;
esac
done
if test -f "/usr/share/man/man1/$name.1.gz"
then
less /usr/share/man/man1/"$name".1.gz
else
echo "There isn't information about this comma
fi
```

рис. 3

```
agsargsyan@agsargsyan-VirtualBox:~/lab13$ chmod u+x
agsargsyan@agsargsyan-VirtualBox:~/lab13$ ./sch2.sh
There isn't information about this command
agsargsyan@agsargsyan-VirtualBox:~/lab13$
```

рис. 4

- Используя встроенную переменную \$RANDOM, написали командный файл, генерирующий случайную последовательность букв латинского алфавита.(Рис.5-6)



рис. 5

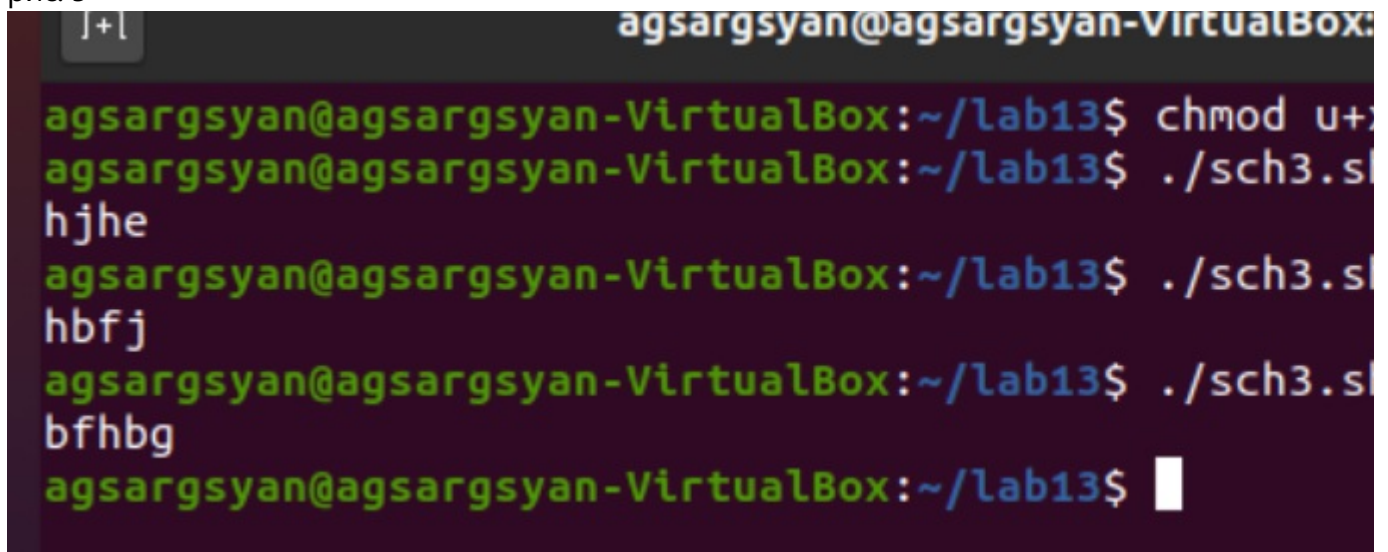


рис. 6

**ВЫВОД:** В ходе лабораторной работы я изучил основы программирования в оболочке ОС UNIX. Научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.