

# **Лабораторная работа № 12**

**Операционные системы**

Голованова Мария Константиновна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
<b>4</b>	<b>Выводы</b>	<b>11</b>
	<b>Список литературы</b>	<b>12</b>

## Список иллюстраций

3.1	Программа №1 . . . . .	8
3.2	Программа №2 . . . . .	9
3.3	Программа №3 . . . . .	10

## Список таблиц

# 1 Цель работы

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

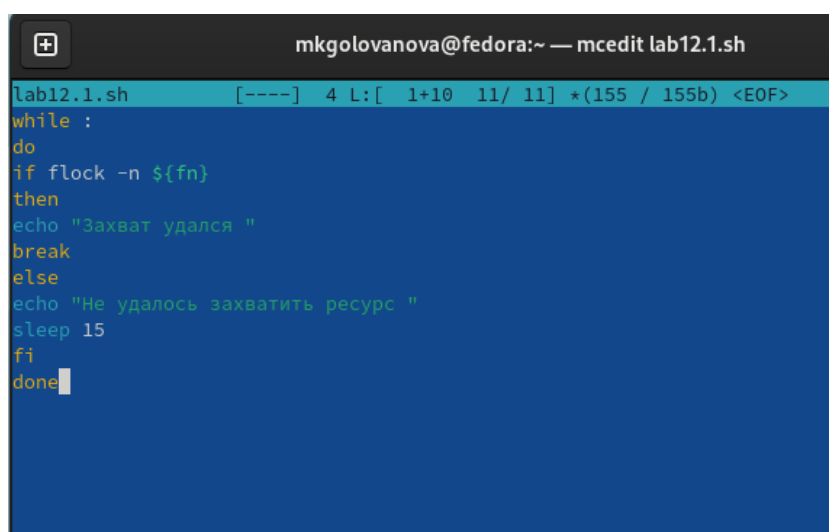
## 2 Задание

1. Написать 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 ресурс используется соответствующим командным файлом (процессом). Запустить 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. Доработать программу так, чтобы имелась возможность взаимодействия трёх и более процессов.
2. Реализовать команду `man` с помощью командного файла. Изучите содержимое каталога `/usr/share/man/man1`. В нем находятся архивы текстовых файлов, содержащих справку по большинству установленных в системе программ и команд. Каждый архив можно открыть командой `less` сразу же просмотрев содержимое справки. Командный файл должен получать в виде аргумента командной строки название команды и в виде результата выдавать справку об этой команде или сообщение об отсутствии справки, если соответствующего файла нет в каталоге `man1`.
3. Используя встроенную переменную `$RANDOM`, напишите командный файл, генерирующий случайную последовательность букв латинского алфавита. Учтите, что `$RANDOM` выдаёт псевдослучайные числа в диапазоне от 0 до

32767.

### 3 Выполнение лабораторной работы

1. Я написала командный файл, реализующий упрощённый механизм семафоров, и запустила его 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. Затем я доработала программу так, чтобы имелась возможность взаимодействия трёх и более процессов (рис. 3.1).



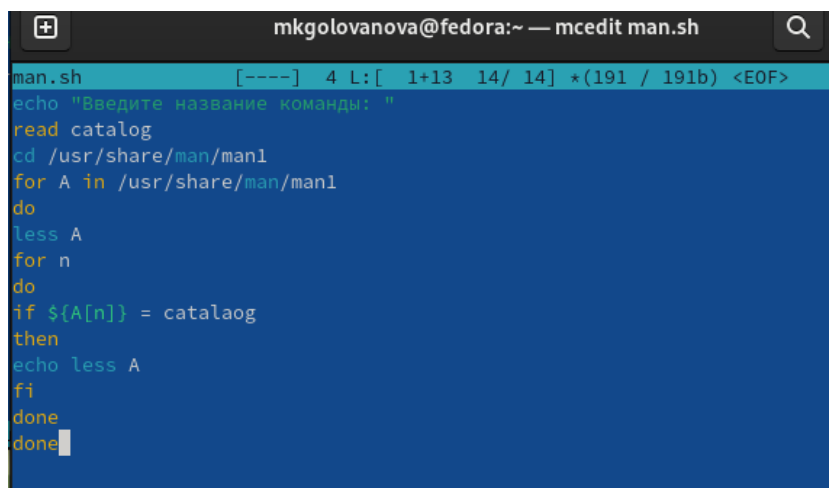
```
mkgolovanova@fedora:~ — mcedit lab12.1.sh
lab12.1.sh [----] 4 L:[ 1+10 11/ 11] *(155 / 155b) <EOF>
while :
do
if flock -n ${fn}
then
echo "Захват удался "
break
else
echo "Не удалось захватить ресурс "
sleep 15
fi
done
```

Рис. 3.1: Программа №1

2. Я реализовала команду man с помощью командного файла, изучив содержимое каталога /usr/share/man/man1. В нем находятся архивы текстовых файлов, содержащих справку по большинству установленных в системе



программ и команд. Каждый архив можно открыть командой `less` сразу же просмотрев содержимое справки. Командный файл получает в виде аргумента командной строки название команды и в виде результата выдаёт справку об этой команде или сообщение об отсутствии справки, если соответствующего файла нет в каталоге `man1` (рис. 3.2).



```
man.sh [----] 4 L: [ 1+13 14/ 14] *(191 / 191b) <EOF>
echo "Введите название команды: "
read catalog
cd /usr/share/man/man1
for A in /usr/share/man/man1
do
less A
for n
do
if ${A[n]} = catalaog
then
echo less A
fi
done
done
```

Рис. 3.2: Программа №2

3. Используя встроенную переменную `$RANDOM`, Я написала командный файл, генерирующий случайную последовательность букв латинского алфавита. (рис. 3.3).

```
Открыть ▾ + • random.sh ~/
#!/bin/bash
Denominations="a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q
r
s
t
u
v
w
x
y
z"
denomination=(${Denominations})
number=${#denomination[$RANDOM%32767+1]}
echo -n "${denomination[$((RANDOM%number%32767+1))]} "
```

Рис. 3.3: Программа №3

## 4 Выводы

Я научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

## **Список литературы**