

# **Отчет по лабораторной работе №14**

**Дисциплина:Операционные системы**

Абрамова Ульяна Михайловна

# 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>
3.1	Задание 1 . . . . .	8
3.2	Задание 2 . . . . .	10
3.3	Задание 3 . . . . .	10
<b>4</b>	<b>Выводы</b>	<b>12</b>
	<b>Список литературы</b>	<b>13</b>

# Список иллюстраций

3.1	1	.....	9
3.2	2	.....	9
3.3	3	.....	10
3.4	4	.....	10
3.5	5	.....	11
3.6	6	.....	11

## **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

### 3.1 Задание 1

Написали командный файл, реализующий упрощённый механизм семафоров. Командный файл в течение некоторого времени  $t_1$  дожидается освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использует его в течение некоторого времени  $t_2 < t_1$ , также выдавая информацию о том, что ресурс используется 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. 3.1, 3.2).



```
gedit lab14_1.sh

Открыть ▼ +

1 #!/bin/bash
2 while test -f lockfile
3 do
4 sleep 1
5 echo "жду разблокировки файла"
6 done
7 touch lockfile
8 let c=10
9 while ((c-=1))
10 do
11 echo "Пишу в файл..."
12 echo "Записываю в файл...">>lockfile
13 sleep 1
14 done
15 rm lockfile
```

Рис. 3.1: 1

```
[umabramova@umabramova work]$ chmod +x lab14_1.sh
[umabramova@umabramova work]$ ./lab14_1.sh
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
[umabramova@umabramova work]$ |
```

Рис. 3.2: 2

## 3.2 Задание 2

Реализовали команду `man` с помощью командного файла. Изучили 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 выдает справку об этой команде или сообщение об отсутствии справки, если соответствующего файла нет в каталоге `man1` (рис. 3.3, 3.4).



Рис. 3.3: 3

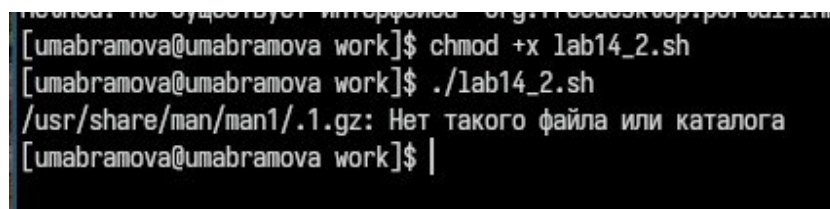
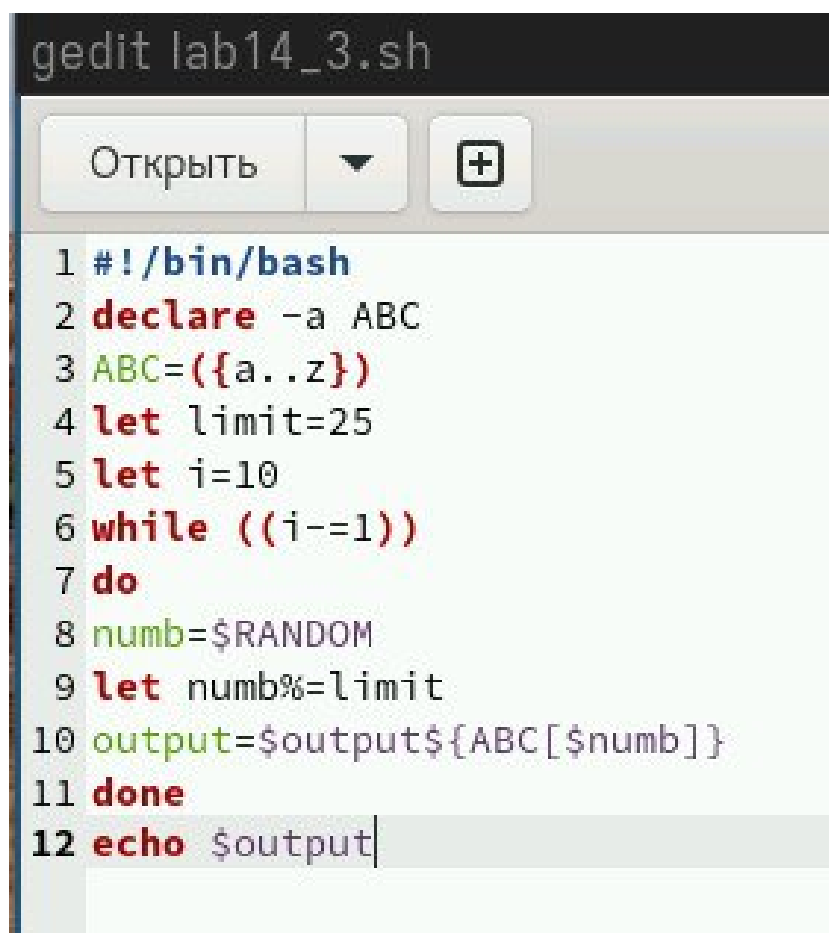


Рис. 3.4: 4

## 3.3 Задание 3

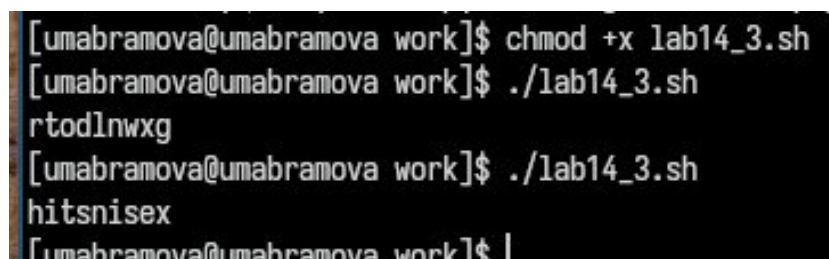
Используя встроенную переменную `$RANDOM`, написали командный файл, генерирующий случайную последовательность букв латинского алфавита (рис. 3.5, 3.6).



```
gedit lab14_3.sh

1 #!/bin/bash
2 declare -a ABC
3 ABC=({a..z})
4 let limit=25
5 let i=10
6 while ((i--=1))
7 do
8 numb=$RANDOM
9 let numb%=limit
10 output=$output${ABC[$numb]}
11 done
12 echo $output
```

Рис. 3.5: 5



```
[umabramova@umabramova work]$ chmod +x lab14_3.sh
[umabramova@umabramova work]$ ./lab14_3.sh
rtodlnwxg
[umabramova@umabramova work]$ ./lab14_3.sh
hitsnisex
[umabramova@umabramova work]$ |
```

Рис. 3.6: 6

## **4 Выводы**

В данной работе я изучила основы программирования в оболочке ОС UNIX. Научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

# **Список литературы**

## **1. Операционные системы**