

# Перезентация по Лабораторной работе 14

Программирование в командном процессоре ОС UNIX. Расширенное программирование”

---

Неустроева И.Н.

2 мая 2024

Российский университет дружбы народов, Москва, Россия

# Информация

---

- Неустроева Ирина Николаевна
- студентка группы НБИ 02-23
- Российский университет дружбы народов
- <https://inneustroeva.github.io/ru/>

# Вводная часть

---

Умение программировать в командном процессоре ОС UNIX - очень важный навык для специалиста в области информационных технологий

# Цели

---

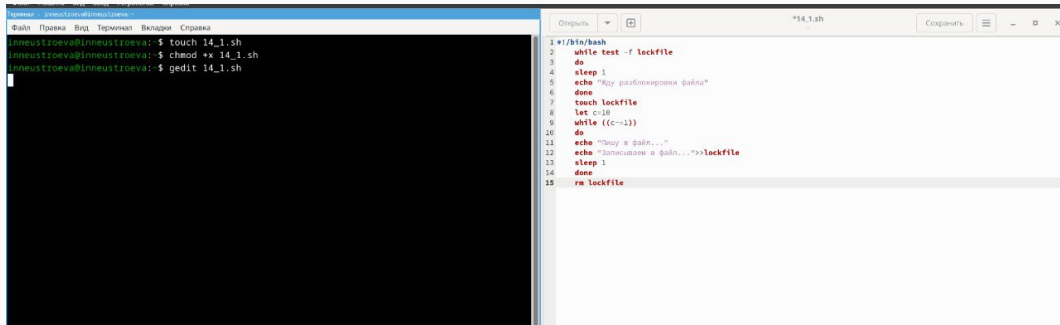
Целью моей работы было: Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

## Основная часть

---



# Командный файл, который реализует упрощенный механизм семафоров



The image shows a terminal window on the left and a text editor window on the right. The terminal window displays the following commands and output:

```
inneustroeva@inneustroeva:~$ touch 14_1.sh
inneustroeva@inneustroeva:~$ chmod +x 14_1.sh
inneustroeva@inneustroeva:~$ gedit 14_1.sh
```

The text editor window shows the content of the file `14_1.sh`:

```
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 "Записываем в файл...">>lockfile
13     sleep 1
14 done
15 rm lockfile
```

```
inneustroeva@inneustroeva:~$ bash 14_1.sh
```

```
Пишу в файл...
```

```
Пишу в файл...
```

```
Пишу в файл...
```

```
Пишу в файл...
```

```
Пишу в файл...
```

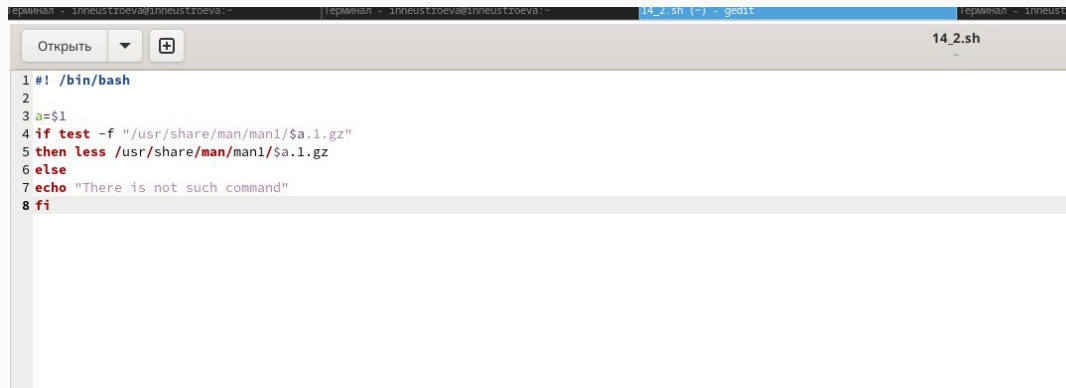
```
Пишу в файл...
```

```
Пишу в файл...
```

```
Пишу в файл...
```

# Реализовать команду man с помощью командного файла

man- команда отвечает за справку по команде



The screenshot shows a terminal window with a title bar containing three tabs: "терминал - inneustroeva@inneustroeva:", "14\_2.sh (~) - gedit", and "терминал - inneust". The active tab is "14\_2.sh (~) - gedit". Below the title bar is a toolbar with a button labeled "Открыть", a dropdown arrow, and a "+" button. The main area of the terminal displays a shell script for the 'man' command. The script is as follows:

```
1 #! /bin/bash
2
3 a=$1
4 if test -f "/usr/share/man/man1/$a.1.gz"
5 then less /usr/share/man/man1/$a.1.gz
6 else
7 echo "There is not such command"
8 fi
```

## Реализация команды `map` при помощи командного файла

```
inneustroeva@inneustroeva:~$ touch 14_2.sh  
inneustroeva@inneustroeva:~$ chmod +x 14_2.sh  
inneustroeva@inneustroeva:~$ gedit 14_2.sh  
inneustroeva@inneustroeva:~$ ./14_2.sh mkdir
```

1

# Справка по команде mkdir

```
Файл  Правка  Вид  Терминал  Вкладки  Справка
inneustroeva@inneustroeva:~
ESC[4mMKDIRESC[24m(1)
ESC[4mMKDIRESC[24m(1)
ESC[1mNAMEESC[0m
mkdir - make directories
ESC[1mSYNOPSISESC[0m
ESC[1mmkdir ESC[22m[ESC[4mOPTIONESC[24m]... ESC[4mDIRECTORYESC[24m...
ESC[1mDESCRIPTIONESC[0m
Create the DIRECTORY(ies), if they do not already exist.

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

ESC[1m-mESC[22m, ESC[1m--modeESC[22m=ESC[4mMODEESC[0m
set file mode (as in chmod), not a=rwx - umask

ESC[1m-pESC[22m, ESC[1m--parentsESC[0m
no error if existing, make parent directories as needed, with their file modes unaffected by any ESC[1m-m ESC[22moption.

ESC[1m-vESC[22m, ESC[1m--verboseESC[0m
print a message for each created directory

ESC[1m-Z ESC[22mset SELinux security context of each created directory to the default type

ESC[1m--contextESC[22m[=ESC[4mCTXESC[24m]
like ESC[1m-ZESC[22m, or if CTX is specified then set the SELinux or SMACK security context to CTX

ESC[1m--help ESC[22mdisplay this help and exit

ESC[1m--versionESC[0m
output version information and exit

ESC[1mAUTHORESC[0m
```

# Командный файл, генерирующий случайную последовательность букв латинского алфавита

```
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
```

## Реализация алгоритма задания 3

```
inneustroeva@inneustroeva:~$ touch 14_3.sh
inneustroeva@inneustroeva:~$ chmod +x 14_3.sh
inneustroeva@inneustroeva:~$ gedit 14_3.sh
inneustroeva@inneustroeva:~$ ./14_3.sh
krydccucp
inneustroeva@inneustroeva:~$ ./14_3.sh
fxqheboyh
inneustroeva@inneustroeva:~$ ./14_3.sh
```

**Заключительная часть.**

---



В данной работе я научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Навыки, приобретенные мною в ходе выполнения данной лабораторной работы, пригодятся мне в будущей профессии.

... ..