

# Программирование в командном процессоре ОС UNIX.

---

Булат Исаев<sup>1</sup>

13 июня, 2023, Москва, Россия

<sup>1</sup>Российский Университет Дружбы Народов

# Цели и задачи работы

---

## Цель лабораторной работы

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

# Задачи лабораторной работы

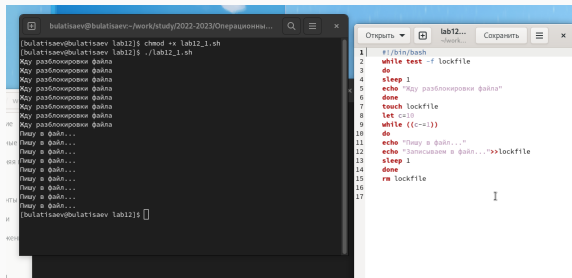
1 Выполнить 3 задания

# **Процесс выполнения лабораторной работы**

---

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).

# Выполнение работы



The image shows a terminal window on the left and a script editor on the right. The terminal window displays the execution of a script named `lab12_1.sh` which attempts to write to a file named `lockfile`. The first few attempts succeed, but then the process is blocked by a message "Пишу в файл..." (Writing to file...). The script editor on the right shows the code for `lab12_1.sh`, which is a bash script that creates a while loop to write to `lockfile` until it is removed.

```
[bulatisev@bulatisev ~/work/study/2022-2023/Операционны...]
[bulatisev@bulatisev lab12]$ chmod +x lab12_1.sh
[bulatisev@bulatisev lab12]$ ./lab12_1.sh
Кду разблокировки файла
Кду разблокировки файла
Кду разблокировки файла
Кду разблокировки файла
Кду разблокировки файла
Кду разблокировки файла
Кду разблокировки файла
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
[bulatisev@bulatisev lab12]$
```

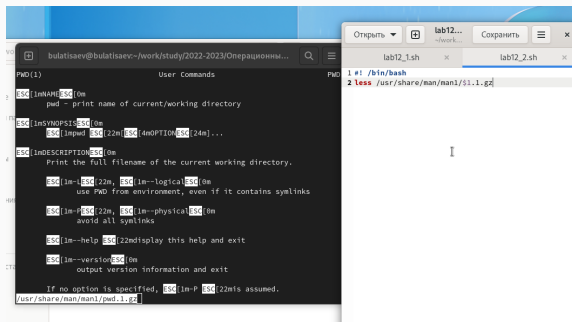
```
#!/bin/bash
1 while test -f lockfile
2 do
3   sleep 1
4   echo "Кду разблокировки файла"
5 done
6 touch lockfile
7 let c+=1
8 while ((c>=1))
9 do
10  echo "Пишу в файл..."
11  echo "Записываем в файл...">>lockfile
12  sleep 1
13 done
14 rm lockfile
15
16
17
```

Рис. 1: Задание 1

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.



# Выполнение работы



The image shows two overlapping terminal windows. The background window is titled 'bulatisaev@bulatisaev:~/work/study/2022-2023/Операционны...' and displays the 'User Commands' for 'pwd'. The foreground window is titled 'lab12...' and shows the command 'less /usr/share/man/man1/pwd.1.gz' being executed.

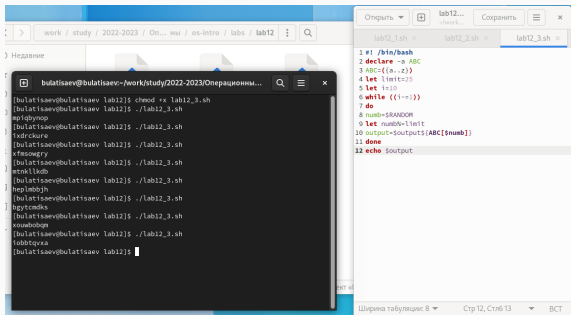
```
bulatisaev@bulatisaev:~/work/study/2022-2023/Операционны...
PMD(1) User Commands
1  [ESC]lnNAME[ESC](0m
2  pwd - print name of current/working directory
10 [ESC]lnSYNOPSIS[ESC](0m
11 [ESC]lnpwd[ESC](22m[ESC](4nOPTION[ESC](24m)...
4 [ESC]lnDESCRIPTION[ESC](0m
5  Print the full filename of the current working directory.
6  [ESC]ln-[ESC](22m, [ESC]ln--logical[ESC](0m
7  use PWD from environment, even if it contains symlinks
8  [ESC]ln-P[ESC](22m, [ESC]ln--physical[ESC](0m
9  avoid all symlinks
10 [ESC]ln--help[ESC](22m;display this help and exit
11 [ESC]ln--version[ESC](0m
12 output version information and exit
13
14 If no option is specified, [ESC]ln-P[ESC](22m is assumed.
15 /usr/share/man/man1/pwd.1.gz
```

```
lab12_1.sh lab12_2.sh
1 #! /bin/bash
2 less /usr/share/man/man1/pwd.1.gz
```

Рис. 2: Задание 2

3. Используя встроенную переменную `$RANDOM`, написали командный файл, генерирующий случайную последовательность букв латинского алфавита

# Выполнение работы



The image shows a screenshot of a computer screen with two windows. The background window is a terminal with a black background and white text. It shows a user named 'bulatissaev' at a prompt, running a series of commands to execute a script named 'lab12\_3.sh'. The commands are: 'chmod +x lab12\_3.sh', followed by several './lab12\_3.sh' commands. The foreground window is a code editor with a light blue header bar. It has tabs for 'lab12\_1.sh', 'lab12\_2.sh', and 'lab12\_3.sh'. The 'lab12\_3.sh' tab is active, showing a shell script. The script starts with '#!/bin/bash', followed by 'declare -a ABC', 'ABC=({a..z})', 'let limit=25', 'let i=0', a 'while ((i<=limit))' loop, 'do', 'numb=\$((i+1))', 'let numb=\$((numb+1))', 'output=\$(cat /dev/urandom | tr -dc 'a-z' | fold -w 25 | xargs echo | sha256sum | cut -d ' ' -f 1)', 'done', and 'echo \$output'.

```
bulatissaev@bulatissaev:~/work/study/2022-2023/Операционны...  
[bulatissaev@bulatissaev lab12]$ chmod +x lab12_3.sh  
[bulatissaev@bulatissaev lab12]$ ./lab12_3.sh  
apigbyndp  
[bulatissaev@bulatissaev lab12]$ ./lab12_3.sh  
fxdrckure  
[bulatissaev@bulatissaev lab12]$ ./lab12_3.sh  
xfssowgry  
[bulatissaev@bulatissaev lab12]$ ./lab12_3.sh  
atnkllkdb  
[bulatissaev@bulatissaev lab12]$ ./lab12_3.sh  
heplmbgh  
[bulatissaev@bulatissaev lab12]$ ./lab12_3.sh  
bgytmdks  
[bulatissaev@bulatissaev lab12]$ ./lab12_3.sh  
xoumbobgn  
[bulatissaev@bulatissaev lab12]$ ./lab12_3.sh  
iobbtqvxa  
[bulatissaev@bulatissaev lab12]$
```

```
#!/bin/bash  
2 declare -a ABC  
3 ABC=({a..z})  
4 let limit=25  
5 let i=0  
6 while ((i<=limit))  
7 do  
8   numb=$((i+1))  
9   let numb=$((numb+1))  
10  output=$(cat /dev/urandom | tr -dc 'a-z' | fold -w 25 | xargs echo | sha256sum | cut -d ' ' -f 1)  
11 done  
12 echo $output
```

Рис. 3: Задание 3

## **Выводы по проделанной работе**

---

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