

# Лабораторная работа № 14

Операционные системы

---

Иванов Сергей Владимирович, НПИбд-01-23

4 мая 2024

Российский университет дружбы народов, Москва, Россия

Цель данной лабораторной работы - изучить основы программирования в оболочке ОС UNIX, научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

## **Выполнение работы**

---

## Создание и исполнение файла

Создаю файл для первой программы, пишу ее, проверяю работу

```
[svivanov1@svivanov1 ~]$ touch 121.sh
[svivanov1@svivanov1 ~]$ chmod +x 121.sh
[svivanov1@svivanov1 ~]$ bash 121.sh
File is blocked
File is unlocked
File is blocked
File is unlocked
File is blocked
File is unlocked
```

**Рис. 1:** Создание и исполнение файла

# Код программы

Командный файл, реализующий упрощённый механизм семафоров

```
1 #!/bin/bash
2
3 lockfile="./lock.file"
4 exec {fn}>$lockfile
5
6 while test -f "$lockfile"
7 do
8   if flock -n ${fn}
9   then
10     echo "File is blocked"
11     sleep 5
12     echo "File is unlocked"
13     flock -u ${fn}
14   else
15     echo "File is blocked"
16     sleep 5
```

## Изучение содержимого папки

Реализовываем команду `man` с помощью командного файла

```
zless.1.gz  
zmore.1.gz  
znew.1.gz  
zsoelim.1.gz  
zvbi-atsc-cc.1.gz  
zvbi-chains.1.gz  
zvid.1.gz  
zvbi-ntsc-cc.1.gz  
[svivanov1@svivanov1 ~]$ ls /usr/share/man/man1
```

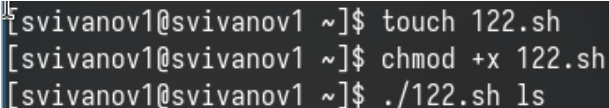
**Рис. 3:** Изучение содержимого папки

Код этой программы

```
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 no such command"  
8 fi
```

Рис. 4: Код программы

Проверяю работу командного файла

A terminal window with a dark background and light gray text. It shows three lines of commands being executed in a shell. The first line creates a file named 122.sh using the 'touch' command. The second line makes the file executable using 'chmod +x'. The third line runs the script './122.sh' followed by the 'ls' command to list the contents of the current directory.

```
[svivanov1@svivanov1 ~]$ touch 122.sh  
[svivanov1@svivanov1 ~]$ chmod +x 122.sh  
[svivanov1@svivanov1 ~]$ ./122.sh ls
```

**Рис. 5:** Исполнение программы



# Результат работы программы

Командный файл работает так же, как и команда `man`

```
ESC[1m-aESC[22m, ESC[1m--allESC[0m
do not ignore entries starting with .

ESC[1m-AESC[22m, ESC[1m--almost-allESC[0m
do not list implied . and ..

ESC[1m--authorESC[0m
with ESC[1m-lESC[22m, print the author of each file

ESC[1m-bESC[22m, ESC[1m--escapeESC[0m
print C-style escapes for nongraphic
characters

ESC[1m--block-sizeESC[22m=ESC[4mSIZEESC[0m
```

Рис. 6: Результат работы программы

## Создание и исполнение файла

Создаю файл для третьей программы, пишу программу и проверяю ее.

```
[svivanov1@svivanov1 ~]$ touch 123.sh  
[svivanov1@svivanov1 ~]$ chmod +x 123.sh  
[svivanov1@svivanov1 ~]$ bash 123.sh 20  
xlpsduaftfpqxlcczdo  
[svivanov1@svivanov1 ~]$
```

**Рис. 7:** Создание и исполнение файла

Код программы, генерирующей случайную последовательность букв латинского алфавита

```
1 #!/bin/bash
2
3 a=$1
4
5 for ((i=0; i<$a; i++))
6 do
7     ((char=$RANDOM%26+1))
8     case $char in
9         1) echo -n a;; 2) echo -n b;; 3) echo -n c;; 4) echo -n d;; 5) echo -n e;; 6) echo -n f;;
10        7) echo -n g;; 8) echo -n h;; 9) echo -n i;; 10) echo -n j;; 11) echo -n k;; 12) echo -n l;;
11        13) echo -n m;; 14) echo -n n;; 15) echo -n o;; 16) echo -n p;; 17) echo -n r;; 18) echo -n s;;
12        19) echo -n t;; 20) echo -n q;; 21) echo -n u;; 22) echo -n v;;
13        23) echo -n w;; 24) echo -n x;; 25) echo -n y;; 26) echo -n z;;
14    esac
15 done
```

Рис. 8: Код программы

## Вывод

---

При выполнении данной лабораторной работы я изучил основы программирования в оболочке ОС UNIX, научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

<https://esystem.rudn.ru/mod/page/view.php?id=1098933>