

# Презентация по лабораторной работе №12

Операционные системы

---

Никулина Ксения Ильинична

27 апреля 2023

Российский университет дружбы народов, Москва, Россия

## Цель работы

---

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

## Задание

---

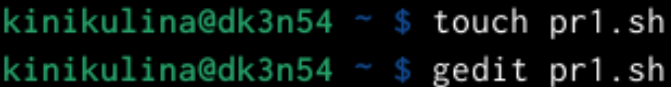
1. Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами:
  - `-i`inputfile — прочитать данные из указанного файла;
  - `-o`outputfile — вывести данные в указанный файл;
  - `-р`шаблон — указать шаблон для поиска;
  - `-C` — различать большие и малые буквы;
  - `-n` — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом `-р`.
2. Написать на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции `exit(n)`, передавая информацию в `о` коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено.

3. Написать командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до N (например 1.tmp, 2.tmp, 3.tmp, 4.tmp и т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют).
4. Написать командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду find).

## Выполнение лабораторной работы

---

1. Создала файл для программы 1

A terminal window with a black background and green text. It shows two commands being executed in a shell. The first command is 'touch pr1.sh' and the second is 'gedit pr1.sh'. Both commands are preceded by the prompt 'kinikulina@dk3n54 ~ \$'.

```
kinikulina@dk3n54 ~ $ touch pr1.sh  
kinikulina@dk3n54 ~ $ gedit pr1.sh
```

Рис. 1: Создание файла



## 2. Написала текст программы 1

```
Открыть ▼ +
1 #!/bin/bash
2 t1=$1
3 t2=$2
4 s1=$(date +%s)
5 s2=$(date +%s)
6 ((t=$s2 - $s1))
7 while ((t<t1))
8 do
9     echo "Ожидание"
10    sleep 1
11    s2=$(date +%s)
12    ((t=$s2 - $s1))
13 done
14 s1=$(date +%s)
15 s2=$(date +%s)
16 ((t=$s2 - $s1))
17 while ((t<t2))
18 do
19     echo "Выполнение"
20    sleep 1
21    s2=$(date +%s)
22    ((t=$s2 - $s1))
23 done
```

### 3. Проверила работу написанной программы

```
kinikulina@dk3n54 ~ $ chmod +x pr1.sh
kinikulina@dk3n54 ~ $ ./pr1.sh 3 5
Ожидание
Ожидание
Ожидание
Выполнение
Выполнение
Выполнение
Выполнение
Выполнение
kinikulina@dk3n54 ~ $
```

Рис. 3: Результат работы программы

## 4. Отредактировала текст

```
1 #!/bin/bash
2
3 function ozhidanie
4 {
5
6
7     s1=$(date +%s")
8
9     s2=$(date +%s")
10
11     ((t=s2-$s1))
12
13     while ((t<t1))
14     do
15
16         echo "Ожидание"
17
18         sleep 1
19
20         s2=$(date +%s")
21
22         ((t=s2-$s1))
23
24         done
25     }
26
27 function vipolnenie
28 {
29
30     s1=$(date +%s")
31
32     s2=$(date +%s")
33
34     ((t=s2-$s1))
35
36     while ((t<t2))
```

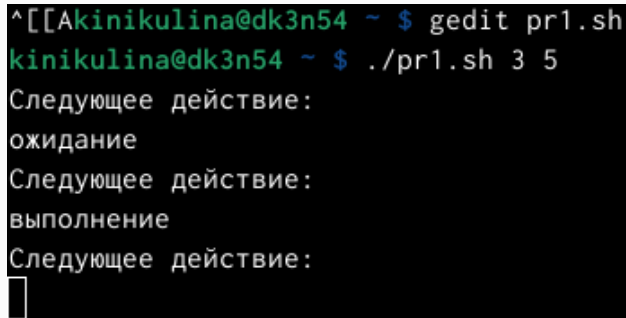
```

7 s2=$(date +%s")
9 ((t=$s2-$s1))
9
1 done
2
3 }
4
5 t1=$1
5
7 t2=$2
3
3 command=$3
3
1 while true
2
3 do
4
5     if [ "$command" == "Выход" ]
5
7     then
3
3     echo "Выход"
3
1     exit 0
2
3     fi
4
5     if [ "$command" == "Ожидание" ]
5
7     then ozhidanie
3
3     fi
3
1     if [ "$command" == "Выполнение" ]
2
3     then vipolnenie
4
5     fi
5
7     echo "Следующее действие: "
3
3     read command
3
1 done

```

Рис. 5: Текст

6. Проверила работу написанной программы

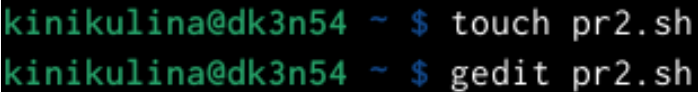
A terminal window with a black background and green text. The prompt is '^[[Akinikulina@dk3n54 ~ \$'. The user enters 'gedit pr1.sh'. The prompt changes to 'kinikulina@dk3n54 ~ \$'. The user enters './pr1.sh 3 5'. The program outputs 'Следующее действие:' followed by 'ожидание' on the next line. It then outputs 'Следующее действие:' followed by 'выполнение' on the next line. Finally, it outputs 'Следующее действие:' followed by a blank line. A blue bar is visible at the bottom of the terminal window.

```
^[[Akinikulina@dk3n54 ~ $ gedit pr1.sh
kinikulina@dk3n54 ~ $ ./pr1.sh 3 5
Следующее действие:
ожидание
Следующее действие:
выполнение
Следующее действие:

```

Рис. 6: Результат работы программы

7. Создала файл для программы 2

A terminal window with a black background and green text. It shows two commands being executed: 'touch pr2.sh' and 'gedit pr2.sh'. The prompt 'kinikulina@dk3n54 ~ \$' is visible for both lines.

```
kinikulina@dk3n54 ~ $ touch pr2.sh  
kinikulina@dk3n54 ~ $ gedit pr2.sh
```

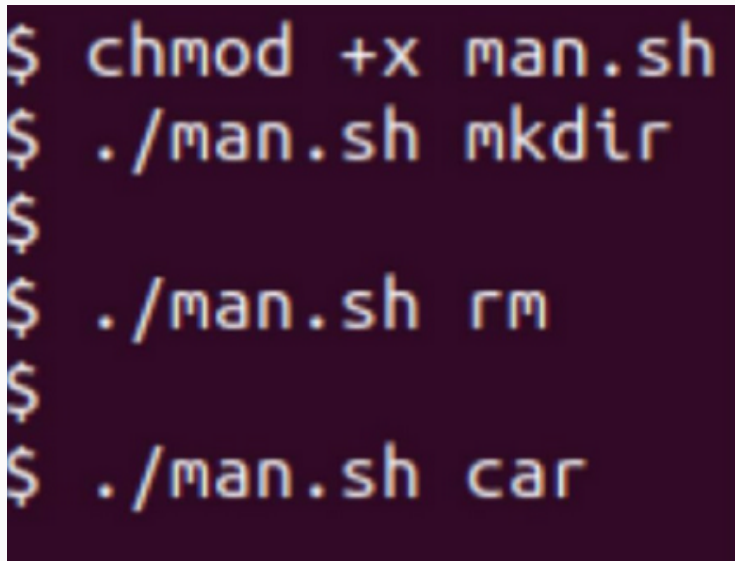
Рис. 7: Создание файла

## 8. Написала текст программы 2

```
1 #!/bin/bash
2
3 a=$1
4
5 if [ -f /usr/share/man/man1/$a.1.gz ]
6
7 then
8
9     gunzip -c /usr/share/man/man1/$1.1.gz | less
10
11 else
12
13     echo "Справки по данной команде нет"
14
15 fi
```

Рис. 8: Создание файла

9. Проверила работу написанной программы

A terminal window with a dark purple background and light blue text. It shows a series of shell commands being entered at a prompt. The commands are: 'chmod +x man.sh', './man.sh mkdir', a blank line, './man.sh rm', a blank line, and './man.sh car'.

```
$ chmod +x man.sh
$ ./man.sh mkdir
$
$ ./man.sh rm
$
$ ./man.sh car
```

Рис. 9: Результат работы программы



10. Создала файл для программы 3

A screenshot of a terminal window with a black background. The prompt 'kinikulina@dk3n54 ~ \$' is shown in green. The command 'touch pr3.sh' is entered in white text, followed by a white cursor box.

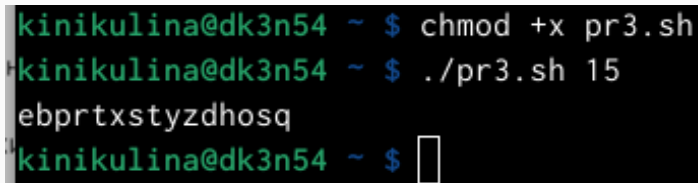
```
kinikulina@dk3n54 ~ $ touch pr3.sh
```

Рис. 10: Создание файла

## 11. Написала текст программы 3

```
1 #!/bin/bash
2
3 a=$1
4
5 for ((i=0; i<$a; i++))
6
7 do
8
9     ((char=$RANDOM%26+1))
10
11     case $char in
12
13         1) echo -n a;; 2) echo -n b;; 3) echo -n c;; 4) echo -n d;; 5) echo -n e;; 6) echo -n
14         f;; 7) echo -n g;; 8) echo -n h;; 9) echo -n i;;
15
16         10) echo -n j;; 11) echo -n k;; 12) echo -n l;; 13) echo -n m;; 14) echo -n n;; 15) echo
17         -n o;; 16) echo -n p;; 17) echo -n q;;
18
19         18) echo -n r;; 19) echo -n s;; 20) echo -n t;; 21) echo -n u;; 22) echo -n v;; 23) echo
20         -n w;; 24) echo -n x;; 25) echo -n y;;
21
22         26) echo -n z;;
23
24     esac
25
26 done
27
28 echo
```

12. Проверила работу написанной программы

A terminal window with a black background and green text. The prompt is 'kinikulina@dk3n54 ~ \$'. The first command is 'chmod +x pr3.sh'. The second command is './pr3.sh 15', which outputs 'ebprtxstyzdhosq'. The third line shows the prompt again with a cursor box.

```
kinikulina@dk3n54 ~ $ chmod +x pr3.sh
kinikulina@dk3n54 ~ $ ./pr3.sh 15
ebprtxstyzdhosq
kinikulina@dk3n54 ~ $
```

Рис. 12: Результат работы программы

## Выводы

---

В ходе выполнения данной лабораторной работы я изучила основы программирования в оболочке ОС UNIX и научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.