

# **Отчёт по лабораторной работе № 13**

**Операционные системы**

Анастасия Романовна Зинченко

# 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>7</b>
<b>4</b>	<b>Выводы</b>	<b>12</b>
	<b>Список литературы</b>	<b>13</b>

## Список иллюстраций

3.1	Программа . . . . .	8
3.2	Проверка . . . . .	8
3.3	Программа . . . . .	9
3.4	Проверка . . . . .	9
3.5	Программа . . . . .	10
3.6	Проверка . . . . .	10
3.7	Программа . . . . .	10
3.8	Проверка . . . . .	11

## **Список таблиц**

# 1 Цель работы

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

## 2 Задание

1. Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами: `-iinputfile` — прочитать данные из указанного файла; `-ooutputfile` — вывести данные в указанный файл; `-rшаблон` — указать шаблон для поиска; `-C` — различать большие и малые буквы; `-n` — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом `-p`.
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  $\infty$  (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 `tag` 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`

### 3 Выполнение лабораторной работы

Используя команды `getopts` `grep`, написала командный файл, который анализирует командную строку с ключами: – `-iinputfile` — прочитать данные из указанного файла; – `-ooutputfile` — вывести данные в указанный файл; – `-rшаблон` — указать шаблон для поиска; – `-C` — различать большие и малые буквы; – `-n` — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом `-r`. (рис. 3.1), (рис. 3.2)

```

1 #! /bin/bash
2
3 while getopts i:o:p:cn optletter
4 do
5     case $optletter in
6         i) iflag=1; ival=$OPTARG;;
7         o) oflag=1; oval=$OPTARG;;
8         p) pflag=1; pval=$OPTARG;;
9         c) cflag=1;;
10        n) nflag=1;;
11        *) echo Illegal option $optletter;;
12    esac
13 done
14
15 if ! test $cflag
16 then
17     cf=-i
18 fi
19
20 if test $nflag
21 then
22     nf=-n
23 fi
24
25 grep $cf $nf $pval $ival >> $oval

```

Рис. 3.1: Программа

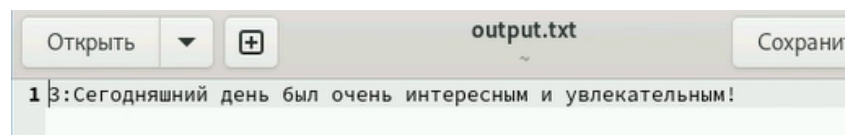


Рис. 3.2: Проверка

Написала на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. (рис. 3.3), (рис. 3.4)



```

1 #! /bin/bash
2 gcc -o cprog 11.2.c
3 ./cprog
4 case $? in
5 0) echo "Число равно нулю";;
6 1) echo "Число больше нуля";;
7 2) echo "Число меньше нуля";;
8 esac

```

Рис. 3.3: Программа

```

[arzinchenko@arzinchenko ~]$ bash 11.2.sh
Введите число: 3
Число больше нуля
[arzinchenko@arzinchenko ~]$ bash 11.2.sh
Введите число: -6
Число меньше нуля
[arzinchenko@arzinchenko ~]$ bash 11.2.sh
Введите число: 0
Число равно нулю

```

Рис. 3.4: Проверка

Написала командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до  $\infty$ . (рис. 3.5), (рис. 3.6)

```

1 #! /bin/bash
2 for ((i=1; i<=$*; i++))
3 do
4 if test -f "$i".tmp
5 then rm "$i".tmp
6 else touch "$i.tmp"
7 fi
8 done

```

Рис. 3.5: Программа

```

[arzzinchenko@arzzinchenko ~]$ bash 11.3.sh 2
[arzzinchenko@arzzinchenko ~]$ ls
1      abc1      bash4.sh      file.txt      monthly      text.txt      Общедоступные
11.1.sh Anastasia bin      git-extended  my_os      work      'Рабочий стол'
11.2.c australia C++      input.txt      output.txt  output.txt  Видео      Шаблоны
11.2.sh backup  conf.txt  #lab07.sh#    output.txt  Документы
11.3.sh bash1.sh cprog     lab07.sh      play       Загрузки
1.tmp  bash2.sh dotfiles  loffile       reports     Изображения
2.tmp  bash3.sh feathers may          ski.places  Музыка
[arzzinchenko@arzzinchenko ~]$ bash 11.3.sh 2
[arzzinchenko@arzzinchenko ~]$ ls
1      australia C++      input.txt      output.txt  Видео      Шаблоны
11.1.sh backup  conf.txt  #lab07.sh#    output.txt  Документы
11.2.c bash1.sh cprog     lab07.sh      play       Загрузки
11.2.sh bash2.sh dotfiles  loffile       reports     Изображения
11.3.sh bash3.sh feathers may          ski.places  Музыка
abc1   bash4.sh file.txt   monthly      text.txt      Общедоступные
Anastasia bin      git-extended my_os      work      'Рабочий стол'

```

Рис. 3.6: Проверка

Написала командный файл, который с помощью команды tar запаковывает в архив все файлы в указанной директории. (рис. 3.7), (рис. 3.8)

```

1 #! /bin/bash
2 find $* -mtime -7 -mtime +0 -type f > FILES.txt
3 tar -cf archive.tar -T FILES.txt

```

Рис. 3.7: Программа

```
[arzinchenko@arzinchenko ~]$ bash 11.4.sh /home/arzinchenko  
find: '/home/arzinchenko/play/games': Отказано в доступе  
tar: Удаляется начальный '/' из имен объектов  
tar: Удаляются начальные '/' из целей жестких ссылок
```

Рис. 3.8: Проверка

## 4 Выводы

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

## **Список литературы**