

Лабораторная работа No 11.

**Программирование в командном процессоре ОС UNIX. Ветвления и
циклы**

Паласиос Фелипе

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	12

Список иллюстраций

3.1	Создание	7
3.2	Результат	8
3.3	Результат	8
3.4	Создание файлов с помощью команды	9
3.5	Удаление файлов с помощью команд	10
3.6	Создание файлов с помощью команды	11

Список таблиц

1 Цель работы

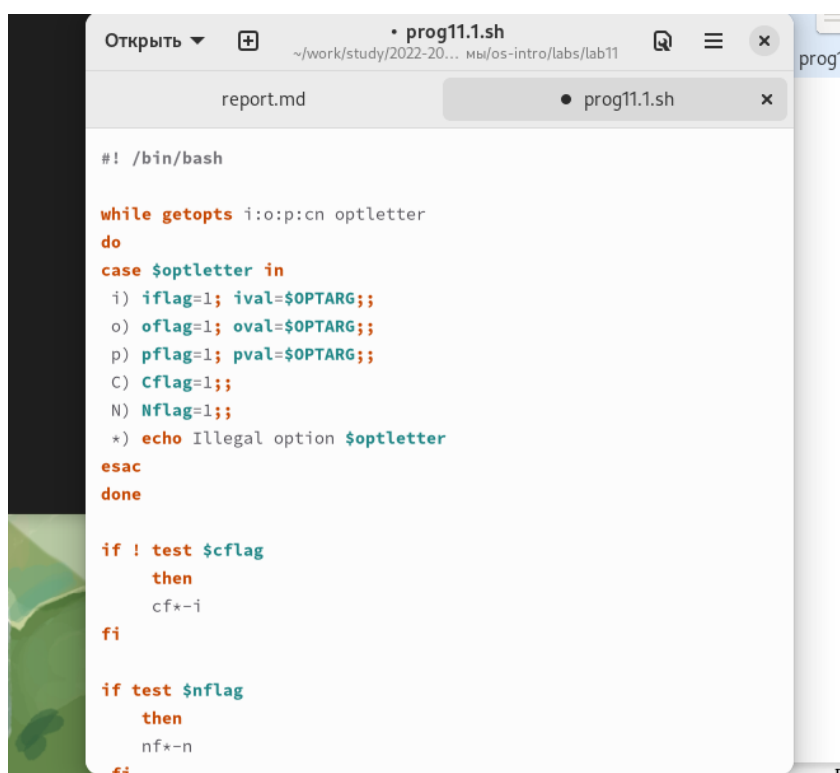
Изучить основы программирования в оболочке ОС UNIX. Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

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; `-r` — 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 `-r`.
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 ∞ (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

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` (рис. 3.1), (рис. 3.2)



```
#!/bin/bash

while getopts i:o:p:cn optletter
do
case $optletter in
i) iflag=1; ival=$OPTARG;;
o) oflag=1; oval=$OPTARG;;
p) pflag=1; pval=$OPTARG;;
C) Cflag=1;;
N) Nflag=1;;
*) echo Illegal option $optletter
esac
done

if ! test $Cflag
then
cf*-i
fi

if test $Nflag
then
nf*-n
fi
```

Рис. 3.1: Создание

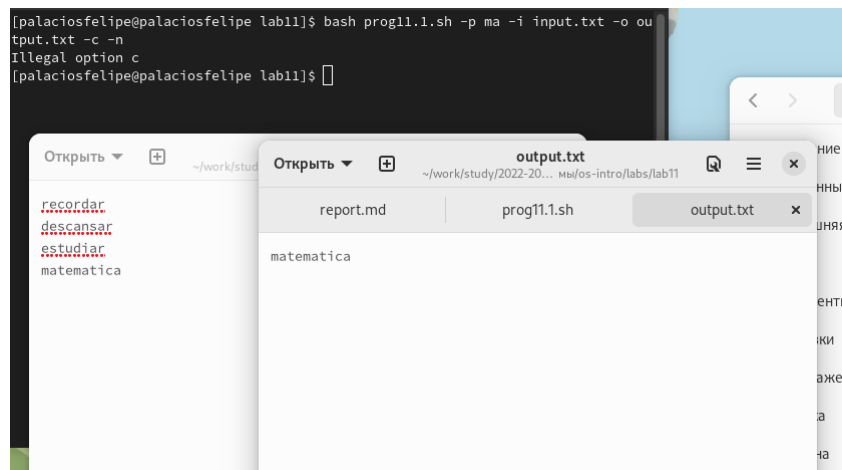


Рис. 3.2: Результат

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 (рис. 3.3)

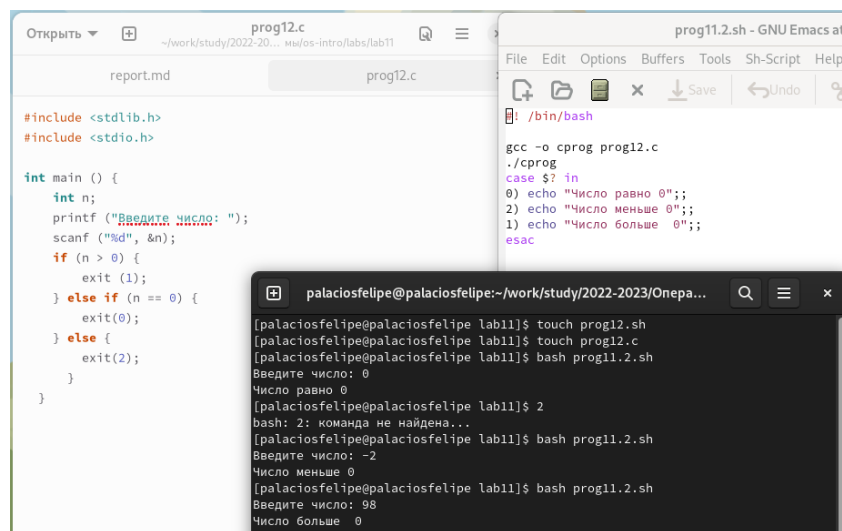


Рис. 3.3: Результат

3. Написать командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до ∞ (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) (рис. 3.4), (рис. 3.5)

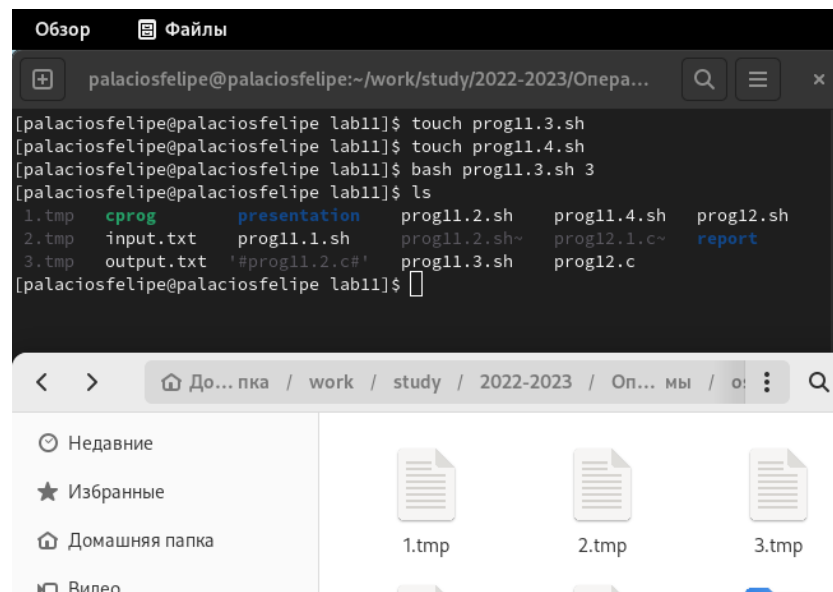
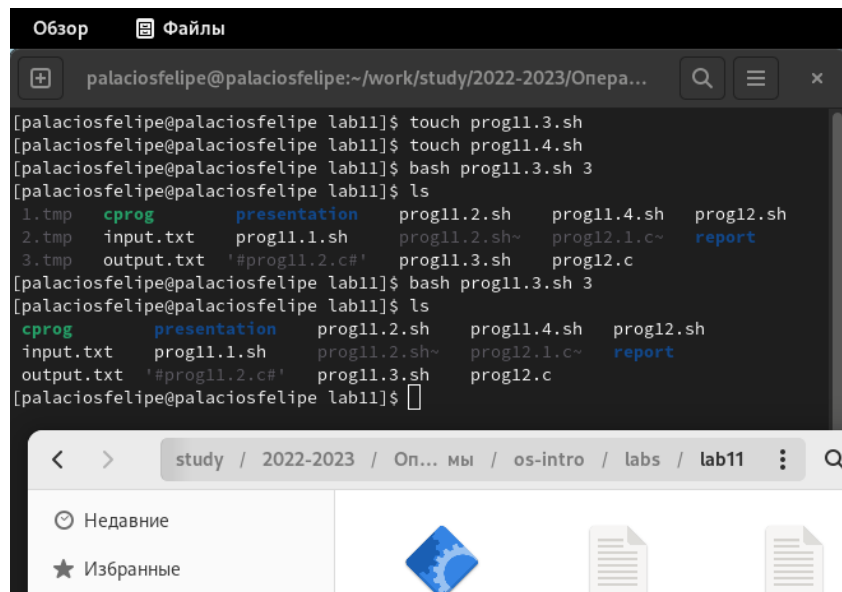


Рис. 3.4: Создание файлов с помощью команды



```
Обзор  📁  Файлы
palaciosfelipe@palaciosfelipe:~/work/study/2022-2023/Onepa...
[palaciosfelipe@palaciosfelipe lab11]$ touch prog11.3.sh
[palaciosfelipe@palaciosfelipe lab11]$ touch prog11.4.sh
[palaciosfelipe@palaciosfelipe lab11]$ bash prog11.3.sh 3
[palaciosfelipe@palaciosfelipe lab11]$ ls
1.tmp  cprog  presentation  prog11.2.sh  prog11.4.sh  prog12.sh
2.tmp  input.txt  prog11.1.sh  prog11.2.sh~  prog12.1.c~  report
3.tmp  output.txt  '#prog11.2.c#'  prog11.3.sh  prog12.c
[palaciosfelipe@palaciosfelipe lab11]$ bash prog11.3.sh 3
[palaciosfelipe@palaciosfelipe lab11]$ ls
cprog  presentation  prog11.2.sh  prog11.4.sh  prog12.sh
input.txt  prog11.1.sh  prog11.2.sh~  prog12.1.c~  report
output.txt  '#prog11.2.c#'  prog11.3.sh  prog12.c
[palaciosfelipe@palaciosfelipe lab11]$
```

study / 2022-2023 / Оп... мы / os-intro / labs / lab11

🕒 Недавние
★ Избранные

Рис. 3.5: Удаление файлов с помощью команд

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`) (рис. 3.6)

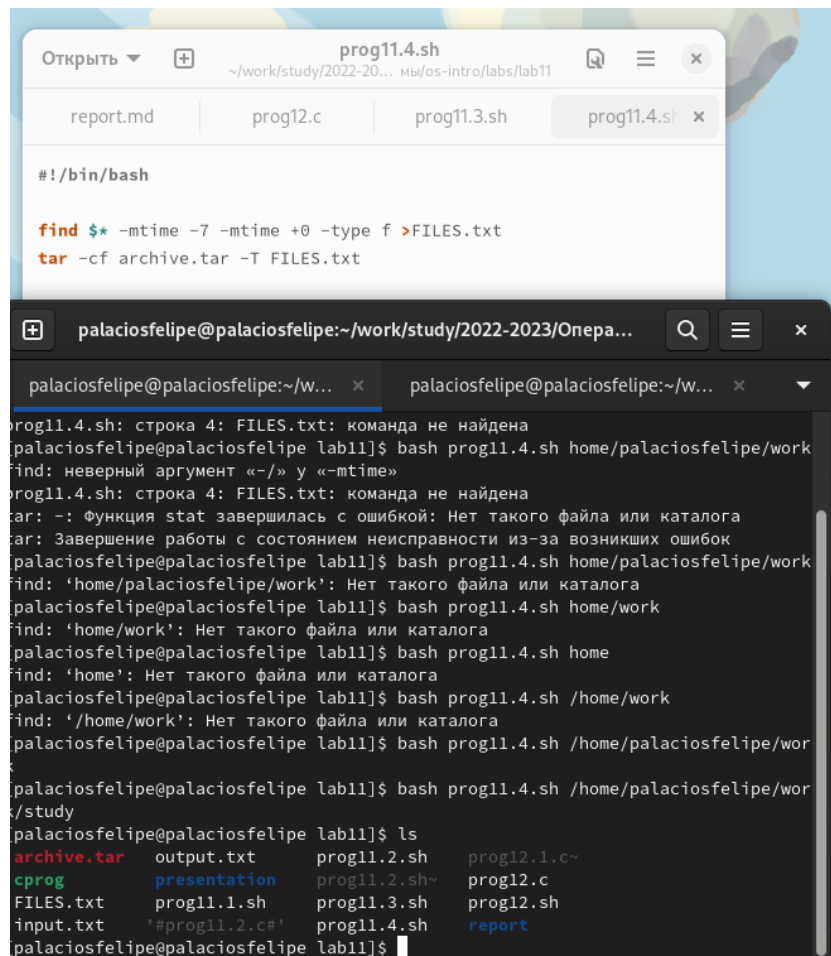


Рис. 3.6: Создание файлов с помощью команды

4 Выводы

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