

Лабораторная работа №11

Операционные системы

Сабралиева М. Н.

Российский университет дружбы народов, Москва, Россия

Информация

- Сабралиева Марворид Нуралиевна
- студентка НБИбд-02-22 кафедры прикладной информатики и теории вероятностей
- Российский университет дружбы народов

```
slide_level: 2  
aspectratio: 169  
section-titles: true  
theme: metropolis
```

- Тема задаётся в файле **Makefile**

```
REVEALJS_THEME = beige
```

Результаты

Элементы презентации

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

1. Используя команды `getopts` `grep`, напишьте командный файл, который анализирует командную строку с ключами: `-i` inputfile — прочитать данные из указанного файла; `-o` outputfile — 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`.

The image shows a terminal window with a light gray title bar. The title bar contains the text "Открыть" (Open) with a dropdown arrow, a plus icon, the filename "lab11.sh", the current directory "~/" (home), and page information "Стр. 1, Поз. 12" (Page 1, Position 12). The terminal content displays a shell script for processing command-line options. The script starts with a shebang, initializes flags, and uses a while loop with getopt to parse options. It sets values for -i, -o, -p, -c, and -n, and then uses grep to search for these values in a file named \$soval. The script ends with a fi statement. Below the main terminal window, there is a smaller, dark-themed terminal window titled "mnsabralieva@fedora:~". It shows the commands "touch lab11.sh" and "chmod +x lab11.sh" being executed.

```
1  #!/bin/bash
2  cflag=0;
3  nflag=0;
4  while getopts 1:o:p:C:n opt
5  do
6  case $opt in
7  1) ival=$OPTARG;;
8  o) oval=$OPTARG;;
9  p) pval=$OPTARG;;
10 C) cflag=1;;
11 n) nflag=1;;
12 esac
13 done
14 if [ $cflag -a $nflag ]
15 then
16 grep -n $pval $ival>$oval
17 elif test $cflag
18 then
19 grep $pval $ival>$oval
20 elif test $nflag
21 then
22 grep -n -1 $pval $ival>$oval
23 else
24 grep -1 $pval $ival>$oval
25 fi
```

mnsabralieva@fedora:~

```
[mnsabralieva@fedora ~]$ touch lab11.sh
[mnsabralieva@fedora ~]$ chmod +x lab11.sh
```

Рис. 1: задание 1

2. Напишем 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.



The image shows a code editor window with a tab labeled 'lab11_2.sh'. The editor displays a shell script with the following content:

```
1 #!/bin/bash
2 gcc -c script2.c
3 gcc -o script2 script2.c
4 ./script2
5 case $? in
6     1) echo положительное;;
7     2) echo равно нулю;;
8     3) echo отрицательное;;
9 esac
```

Рис. 2: задание 2

3. Напишем 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).

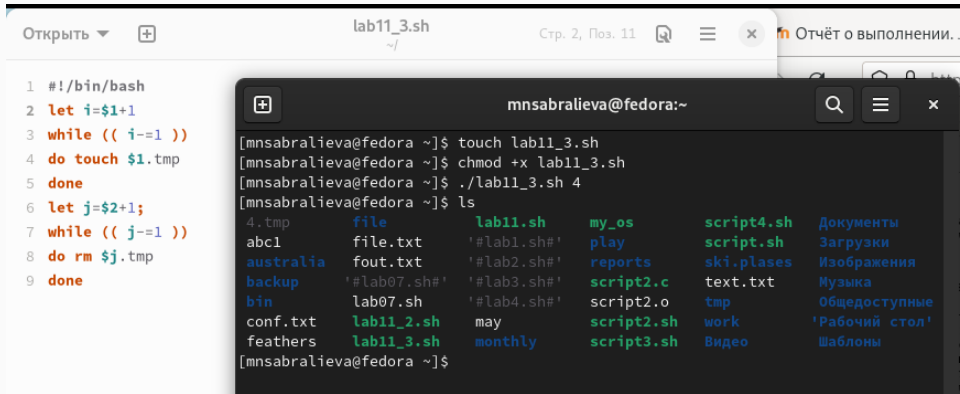


Рис. 3: задание 3

4. Напишем 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`).

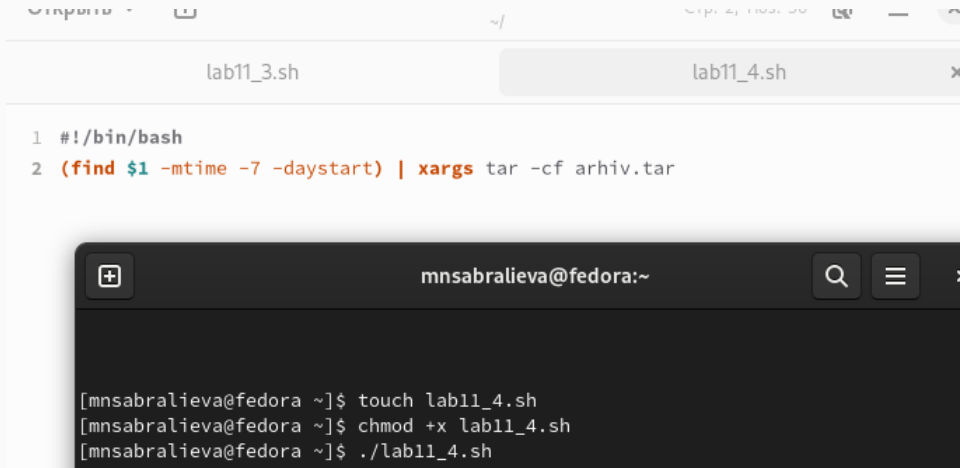


Рис. 4: задание 4

- В данной работе мы изучили основы программирования в оболочке ОС UNIX.
- Также научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

- Запоминается последняя фраза. © Штирлиц

...