

Лабораторная работа №10

Операционные системы

Серёгина Ирина Андреевна

15 апреля 2023

Российский университет дружбы народов, Москва, Россия

Цель работы

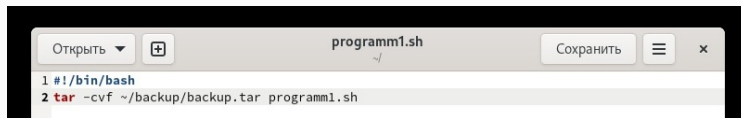
Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.

Задание

1. Написать скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в вашем домашнем каталоге. При этом файл должен архивироваться одним из архиваторов на выбор zip, bzip2 или tar. Способ использования команд архивации необходимо узнать, изучив справку.
2. Написать пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов.
3. Написать командный файл — аналог команды ls (без использования самой этой команды и команды dir). Требуется, чтобы он выдавал информацию о нужном каталоге и выводил информацию о возможностях доступа к файлам этого каталога.
4. Написать командный файл, который получает в качестве аргумента командной строки формат файла (.txt, .doc, .jpg, .pdf и т.д.) и вычисляет количество таких файлов в указанной директории. Путь к директории также передаётся в виде аргумента ко-

Выполнение лабораторной работы

Прописываю первую программу (рис. 1).

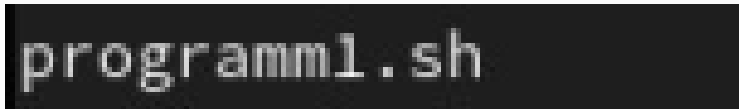


The image shows a code editor window with a title bar containing 'Открыть', a plus icon, 'programm1.sh', 'Сохранить', a hamburger menu icon, and a close icon. The editor area contains two lines of code: '1 #!/bin/bash' and '2 tar -cvf ~/backup/backup.tar programm1.sh'.

```
1 #!/bin/bash
2 tar -cvf ~/backup/backup.tar programm1.sh
```

Figure 1: первая программа

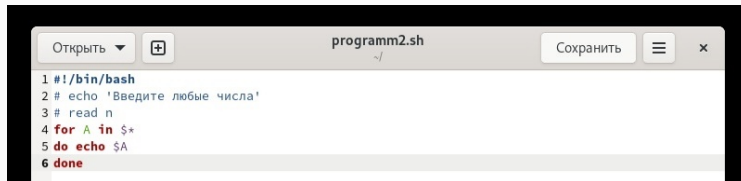
Всё работает (рис. 2).



```
programm1.sh
```

Figure 2: выполнение первой программы

Прописываю вторую программу (рис. 3.)

A screenshot of a code editor window. The title bar shows 'programm2.sh' with a tilde icon. On the left, there is a button 'Открыть' with a dropdown arrow and a '+' icon. On the right, there is a button 'Сохранить', a menu icon (three horizontal lines), and a close icon (an 'x'). The code area contains a shell script with six lines:

```
1 #!/bin/bash
2 # echo 'Введите любые числа'
3 # read n
4 for A in $*
5 do echo $A
6 done
```

Figure 3: вторая программа

Всё работает (рис. 4).

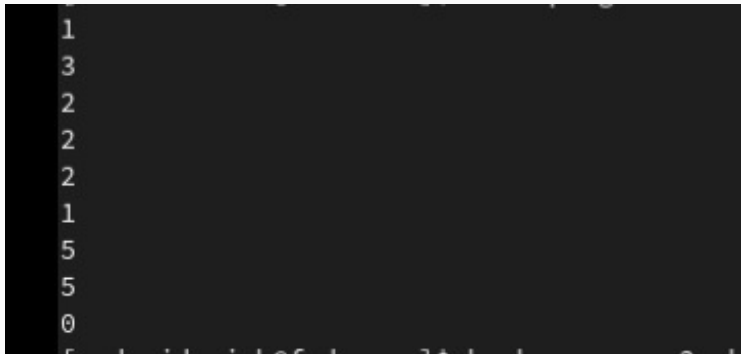


Figure 4: выполнение второй программы

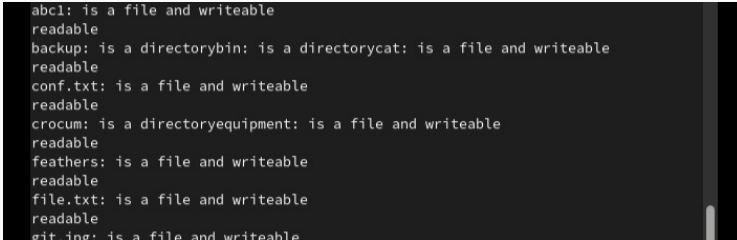
Прописываю третью программу (рис. 5).



```
1 #!/bin/bash
2 for A in *
3 do
4     if test -d "$A"
5     then
6         echo -n "$A: is a directory"
7     else
8         echo -n "$A: is a file and "
9         if test -w $A
10        then
11            echo writeable
12            if test -r $A
13            then
14                echo "readable"
15            else
16                echo "neither readable nor writeable"
17            fi
18        fi
19    fi
20 done
```

Figure 5: третья программа

Всё работает (рис. 6).

A terminal window with a dark background and light gray text. It displays the output of a command that checks permissions for several files and directories. Each entry consists of the name followed by its permissions and a description. The entries are: 'abc1: is a file and writeable readable', 'backup: is a directory', 'bin: is a directory', 'cat: is a file and writeable readable', 'conf.txt: is a file and writeable readable', 'crocum: is a directory', 'equipment: is a file and writeable readable', 'feathers: is a file and writeable readable', 'file.txt: is a file and writeable readable', and 'git.ing: is a file and writeable readable'.

```
abc1: is a file and writeable
readable
backup: is a directory
bin: is a directory
cat: is a file and writeable
readable
conf.txt: is a file and writeable
readable
crocum: is a directory
equipment: is a file and writeable
readable
feathers: is a file and writeable
readable
file.txt: is a file and writeable
readable
git.ing: is a file and writeable
```

Figure 6: выполнение третьей программы

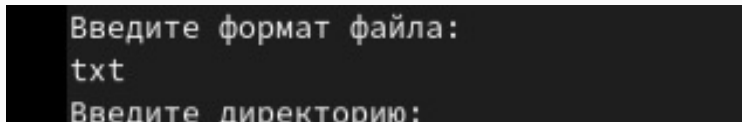
Прописываю четвертую программу (рис. 7).



```
1 #!/bin/bash
2 format=""
3 directory=""
4 echo "Введите формат файла: "
5 read format
6 echo "Введите директорию: "
7 read directory
8 find "${directory}" -name "*.${format}" -type f | wc -l
9 ls
```

Figure 7: четвертая программа

Всё работает (рис. 8).



```
Введите формат файла:  
txt  
Введите директорию:
```

Figure 8: выполнение четвертой программы

Выводы

Я изучила основы программирования в оболочке ОС UNIX/Linux, научилась писать небольшие командные файлы.