

Отчет по лабораторной работе №12

Операционные системы

Стрижов Д. П.

27 апреля 2024

Российский университет дружбы народов, Москва, Россия

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

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 выдал

Выполнение лабораторной работы

Создаю файл `s1.sh`, дай ему доступ для выполнения, чтобы можно было его запускать без ключевого слова `bash`

```
[dpstrizhov@fedora ~]$ touch s1.sh
[dpstrizhov@fedora ~]$ chmod +x s1.sh
[dpstrizhov@fedora ~]$ gedit s1.sh
bash: gedit: команда не найдена...
Установить пакет «gedit», предоставляющий команду «gedit»?
```

Рис. 1: Создание файла для первого скрипта

Прописываю команду для создания резервного файла в каталоге backup, при этом копия архивируется

A terminal window with a light gray background. The first line shows the prompt '1' followed by the command '#!/bin/bash' in blue. The second line shows the prompt '2' followed by the command 'tar -cvf ~/backup/backup.tar s1.sh' in red, with a vertical cursor at the end of the line.

```
1 #!/bin/bash
2 tar -cvf ~/backup/backup.tar s1.sh|
```

Рис. 2: Написание команды для первого скрипта

Создаю файл s2.sh, дай ему доступ для выполнения, чтобы можно было его запускать без ключевого слова bash

```
[dpstrizhov@fedora ~]$ gedit s1.sh  
[dpstrizhov@fedora ~]$ mkdir backup  
[dpstrizhov@fedora ~]$ touch s2.sh  
[dpstrizhov@fedora ~]$ gedit s2.sh  
[dpstrizhov@fedora ~]$ chmod +x s2.sh  
[dpstrizhov@fedora ~]$ gedit s2.sh
```

Рис. 3: Создание файла для второго скрипта


Прописываю скрипт, который выводит произвольное количество переменных

A screenshot of a terminal window with a light gray background. The terminal shows a bash script being written line by line. Line 1 is the shebang `#!/bin/bash`. Line 2 is empty. Line 3 starts a `for` loop with `for A in $*`. Line 4 is the `do` block containing `echo $*`. Line 5 is the `done` keyword, which is currently being typed, with a vertical cursor bar at the end of the word. The lines are numbered 1 through 5 on the left side of the terminal.

```
1 #!/bin/bash
2
3 for A in $*
4     do echo $*
5 done
```

Рис. 4: Написание команды для второго скрипта

Создаю файл `s3.sh`, дай ему доступ для выполнения, чтобы можно было его запускать без ключевого слова `bash`

A terminal window with a dark background and light gray text. It shows three lines of commands and their outputs. The first line is '[dpstrizhov@fedora ~]\$ touch s3.sh'. The second line is '[dpstrizhov@fedora ~]\$ chmod +x s3.sh'. The third line is '[dpstrizhov@fedora ~]\$' followed by a white cursor. A vertical scrollbar is visible on the right side of the terminal window.

```
[dpstrizhov@fedora ~]$ touch s3.sh
[dpstrizhov@fedora ~]$ chmod +x s3.sh
[dpstrizhov@fedora ~]$
```

Рис. 5: Создание файла для третьего скрипта

Прописываю скрипт, который является аналогом ls

```
1 #!/bin/bash
2
3 for A in *
4     if test -d "$A"
5     then
6         echo "$A^ is a diractory"
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 or writeable"
17        fi
18    fi
19 fi
20 done
```

Рис. 6: Написание команды для третьего скрипта

Создаю файл s4.sh, дай ему доступ для выполнения, чтобы можно было его запускать без ключевого слова bash

```
[dpstrizhov@fedora ~]$ touch s4.sh
[dpstrizhov@fedora ~]$ chmod +x s4.sh
bash: chmod: команда не найдена...
Аналогичная команда: 'chmod'
[dpstrizhov@fedora ~]$ chmod +x s4.sh
[dpstrizhov@fedora ~]$
```

Рис. 7: Создание файла для четвертого скрипта

Прописываю скрипт, который вычисляет количество файлов определенного формата в определенной директории

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

Рис. 8: Написание команды для четвертого скрипта

За время выполнения лабораторной работы я изучил основы программирования в оболочке ОС UNIX/Linux. Научился писать небольшие командные файлы.

Лабораторная работа №12: <https://esystem.rudn.ru/course/view.php?id=113>