

Лабораторная работа №12

Операционные системы

Павлова Т. Ю.

Российский университет дружбы народов, Москва, Россия

Целью данной лабораторной работы является изучение основ программирования в оболочке ОС UNIX Linux, а также 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 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

Командный процессор (командная оболочка, интерпретатор команд shell) — это программа, позволяющая пользователю взаимодействовать с операционной системой компьютера. В операционных системах типа UNIX/Linux наиболее часто используются следующие реализации командных оболочек: – оболочка Борна (Bourne shell или sh) — стандартная командная оболочка UNIX/Linux, содержащая базовый, но при этом полный набор функций; – C-оболочка (или csh) — надстройка на оболочкой Борна, использующая C-подобный синтаксис команд с возможностью сохранения истории выполнения команд; – оболочка Корна (или ksh) — напоминает оболочку C, но операторы управления программой совместимы с операторами оболочки Борна; – BASH — сокращение от Bourne Again Shell (опять оболочка Борна), в основе своей совмещает свойства оболочек C и Корна (разработка компании Free Software Foundation). POSIX (Portable Operating System Interface for Computer Environments) — набор стандартов описания интерфейсов взаимодействия операционной системы и прикладных программ. Стандарты POSIX разработаны комитетом IEEE (Institute of Electrical and Electronics Engineers) для обеспечения совместимости различных

Выполнение лабораторной работы

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 (рис. 1), (рис. 2), (рис. 3), (рис. 4).

mc [sanya@sanyaпарав10ва]: ~/Операционные системы

Левая панель	Файл	Команда	Настройки	Правая панель
<	~/Операционные системы/os-intro/labs/lab12/report			. [^]>
'и	Имя	Размер	Дата правки	
/..		-BBERX-	мар	8 00:53
/pandoc		20	мар	8 00:53
/image		50	мар	8 00:53
/bib		16	мар	8 00:53
report.md		5819	мар	8 00:53
lab12-4.sh		0	авг	29 13:18
lab12-3.sh		0	авг	29 13:18
lab12-2.sh		0	авг	29 13:18
lab12-1.sh		0	авг	29 13:18
Makefile		1127	мар	8 00:53

Рис. 1: mc


```
tanya@tatyanaipavlova:~$ cd 'Операционные системы'
tanya@tatyanaipavlova:~/Операционные системы$ cd os-intro
tanya@tatyanaipavlova:~/Операционные системы/os-intro$ cd labs
tanya@tatyanaipavlova:~/Операционные системы/os-intro/labs$ cd lab123
-bash: cd: lab123: Нет такого файла или каталога
tanya@tatyanaipavlova:~/Операционные системы/os-intro/labs$ cd lab12
tanya@tatyanaipavlova:~/Операционные системы/os-intro/labs/lab12$ cd report
tanya@tatyanaipavlova:~/Операционные системы/os-intro/labs/lab12/report$ ls
bib image lab12-1.sh lab12-2.sh lab12-3.sh lab12-4.sh Makefile pandoc report.md
tanya@tatyanaipavlova:~/Операционные системы/os-intro/labs/lab12/report$ chmod +x lab12-1.sh
tanya@tatyanaipavlova:~/Операционные системы/os-intro/labs/lab12/report$ mkdir backup
tanya@tatyanaipavlova:~/Операционные системы/os-intro/labs/lab12/report$ bash lab12-1.sh
tanya@tatyanaipavlova:~/Операционные системы/os-intro/labs/lab12/report$ cd
```

Рис. 2: Команды для компиляции

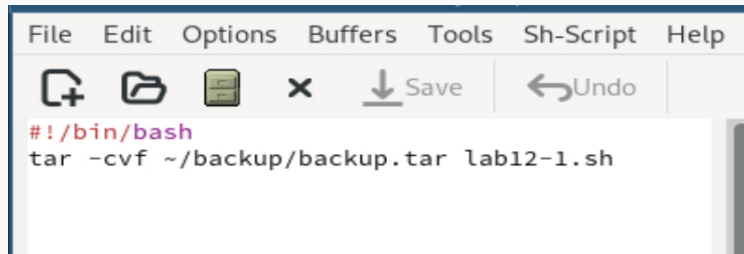


Рис. 3: lab12-1.sh

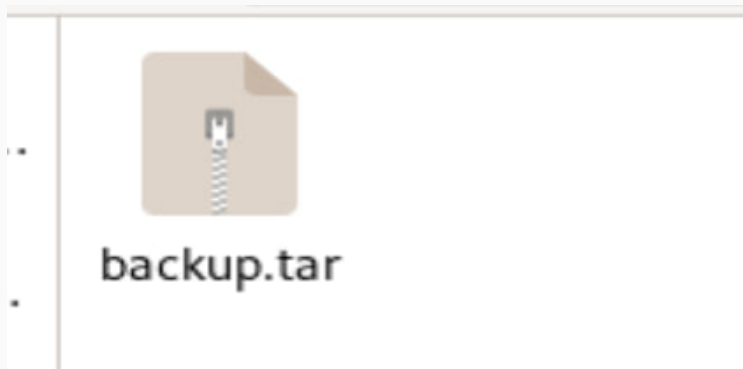
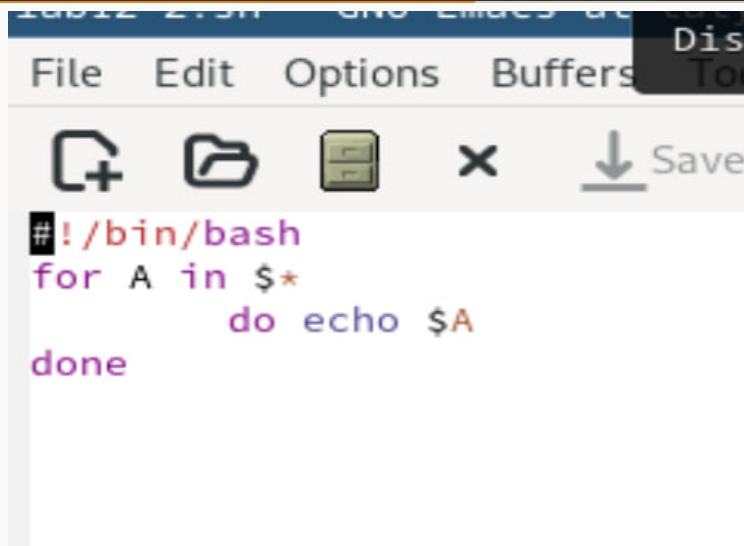


Рис. 4: файл .tar

Задание 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 (рис. 5), (рис. 6).

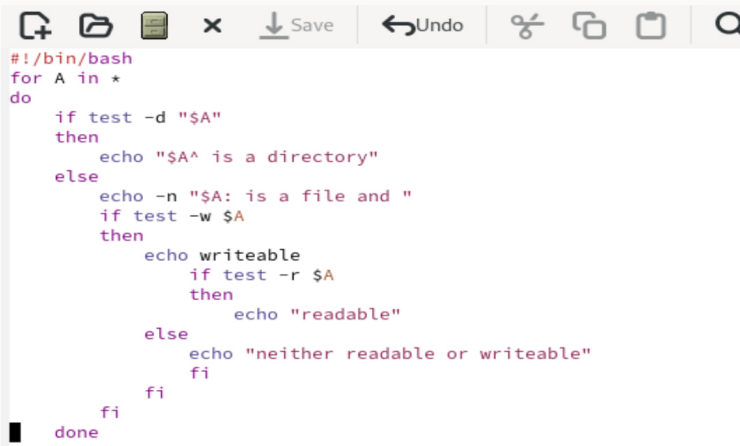


```
tar: Error is not recoverable: exiting now
[tanya@tatanapavlova files_for_lab12]$ bash lab12-2.sh jh oujh iuhjjkj ouihyfpjihj iyt 65 tr5 678
jh
oujh
iuhjjkj
ouihyfpjihj
iyt
65
tr5
678
```

Рис. 6: Команды для компиляции

Задание 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 (рис. 7), (рис. 8).



```
#!/bin/bash
for A in *
do
    if test -d "$A"
    then
        echo "$A^ is a directory"
    else
        echo -n "$A: is a file and "
        if test -w $A
        then
            echo writeable
            if test -r $A
            then
                echo "readable"
            else
                echo "neither readable or writeable"
            fi
        fi
    fi
done
```

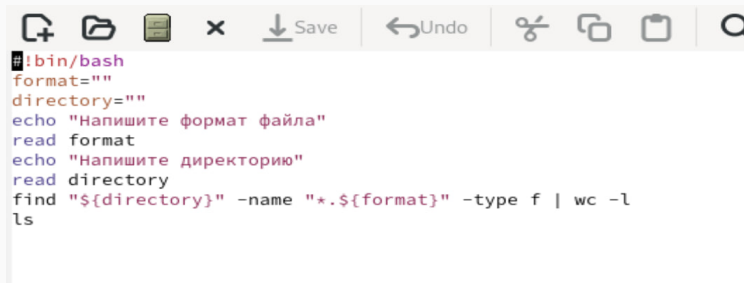
Рис. 7: lab12-3.sh

```
[tanya@tatyana-pavlova files_for_lab12]$ bash lab12-3.sh iloveyou
backup^ is a directory
lab12-1.sh: is a file and writeable
readable
#lab12-2.sh#: is a file and writeable
readable
lab12-2.sh: is a file and writeable
readable
lab12-3.sh: is a file and writeable
readable
lab12-3.sh~: is a file and writeable
readable
lab12-4.sh: is a file and writeable
readable
```

Рис. 8: Команды для компиляции

Задание 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 командной строки (рис. 9), (рис. 10).

A screenshot of a text editor window. The title bar at the top shows icons for a new file, open, save, close, and a search icon. Below the title bar is a toolbar with icons for undo, redo, cut, copy, and paste. The main text area contains a shell script. The script starts with a shebang line, followed by variable declarations for format and directory. It then uses echo to prompt the user for the file format and directory, and read to capture the input. Finally, it uses find to search for files matching the criteria and wc -l to count the number of files found.

```
#!/bin/bash
format=""
directory=""
echo "Напишите формат файла"
read format
echo "Напишите директорию"
read directory
find "${directory}" -name "*.${format}" -type f | wc -l
ls
```

Рис. 9: lab12-4.sh

```
[tanya@tatyana-pavlova ~]$ cd files_for_lab12
[tanya@tatyana-pavlova files_for_lab12]$ chmod +x lab12-4.sh
[tanya@tatyana-pavlova files_for_lab12]$ bash lab12-4.sh
Напишите формат файла
txt
Напишите директорию
/home/tanya
9
  backup lab12-1.sh '#lab12-2.sh#' lab12-2.sh lab12-3.sh lab12-3.sh~ lab12-4.sh lab12-4.sh~
[tanya@tatyana-pavlova files_for_lab12]$
```

Рис. 10: Команды для компиляции

При выполнении данной лабораторной работы, я изучила основы программирования в оболочке ОС UNIX Linux, а также научилась писать небольшие командные файлы.