

Лабораторная работа №10

Дисциплина - Операционные системы

Оширова Юлия Николаевна, НКАбд-02-22

Содержание

1 Цель работы	5
2 Задание	6
3 Теоретическое введение	7
4 Выполнение лабораторной работы	9
5 Выводы	14
Список литературы	15

Список иллюстраций

4.1	Текст первой программы	9
4.2	Проверка выполнения первого задания	10
4.3	Текст второй программы	10
4.4	Проверка работы второй программы	11
4.5	Текст третьей программы	11
4.6	Проверка работы третьей программы	12
4.7	Текст четвертой программы	12
4.8	Проверка работы четвертой программы	13

Список таблиц

1 Цель работы

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

2 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 командной строки.

3 Теоретическое введение

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;

- С-оболочка (или csh) — надстройка на оболочкой Борна, использующая С-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 С 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 UNIX/Linux-подобных операционных систем и переносимости прикладных программ на уровне исходного кода. POSIX-совместимые оболочки разработаны на базе оболочки Корна. Рассмотрим основные элементы программирования в оболочке bash. В других оболочках большинство команд будет совпадать с

описанными ниже. [1]

4 Выполнение лабораторной работы

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.

The screenshot shows a terminal window with the following command history:

```
/home/yn~script01 [----] 52 L:[ 1+ 1 2/ 2] *(82 / 82b) <EOF> [*][X]  
tar -cf script01.tar script01  
mv script01.tar /home/ynoshirova/backup/script01.tar
```

A modal confirmation dialog is displayed in the foreground, asking "Сохранить файл" (Save file) and "Подтверждаете запись файла "/home/ynoshirova/script01"?". The "Сохранить" (Save) button is highlighted with a blue glow, while the "Прервать" (Cancel) button is not.

Рис. 4.1: Текст первой программы

```
ynoshirova@ynoshirova-VirtualBox: ~/backup
'udo sysctl kernel.dmesg_restrict=0 kernel.dmesg_restrict = 0sudo sysctl kernel
.dmesg_restrict=0 kernel.dmesg_restrict = 0'$'\033''0B'$'\033''0B'$'\033''0B'$'
\033''0B'$'\033''0B'$'\033''0B'$'\033''0B'$'\033''0B'$'\033''0B'$'\033''0B'$'\03
3''0B'$'\033''0B'
.wget-hsts
work
Видео
Документы
Загрузки
Изображения
Музыка
Общедоступные
'Рабочий стол'
Шаблоны
ynoshirova@ynoshirova-VirtualBox:~$ ./script01
ynoshirova@ynoshirova-VirtualBox:~$ cd backup/
ynoshirova@ynoshirova-VirtualBox:~/backup$ ls
script01.tar
ynoshirova@ynoshirova-VirtualBox:~/backup$ tar -xf script01.tar
ynoshirova@ynoshirova-VirtualBox:~/backup$ ls
script01  script01.tar
ynoshirova@ynoshirova-VirtualBox:~/backup$ mcedit script01
ynoshirova@ynoshirova-VirtualBox:~/backup$
```

Рис. 4.2: Проверка выполнения первого задания

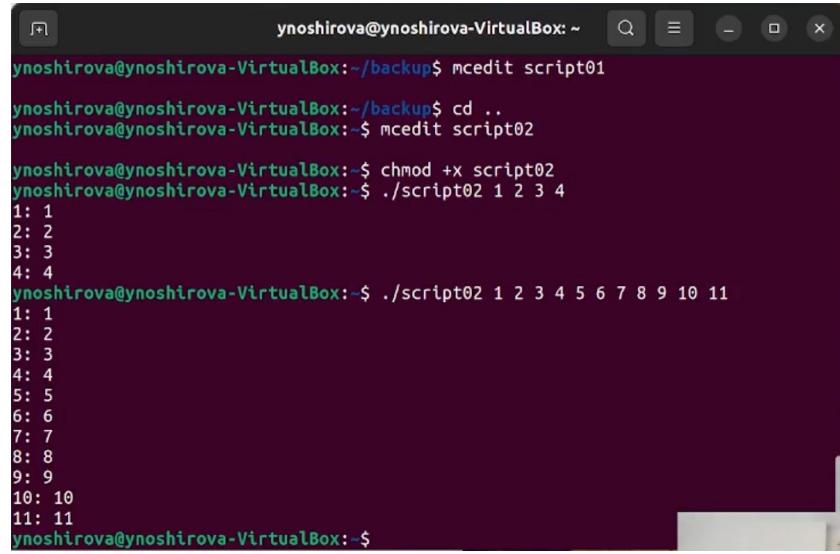
2. Написать пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов.

```
/home/yn~script02 [-M--] 24 L:[ 1+ 4 5/ 6] *(75 / 80b) 0010 0x00A [*][X]
count=0
for parametr in "$@"
do
count=$((count+1))
echo "$count: $parametr"
done
```

Сохранить файл
Подтверждаете запись файла "/home/ynoshirova/script02"?

[Сохранить] [Прервать]

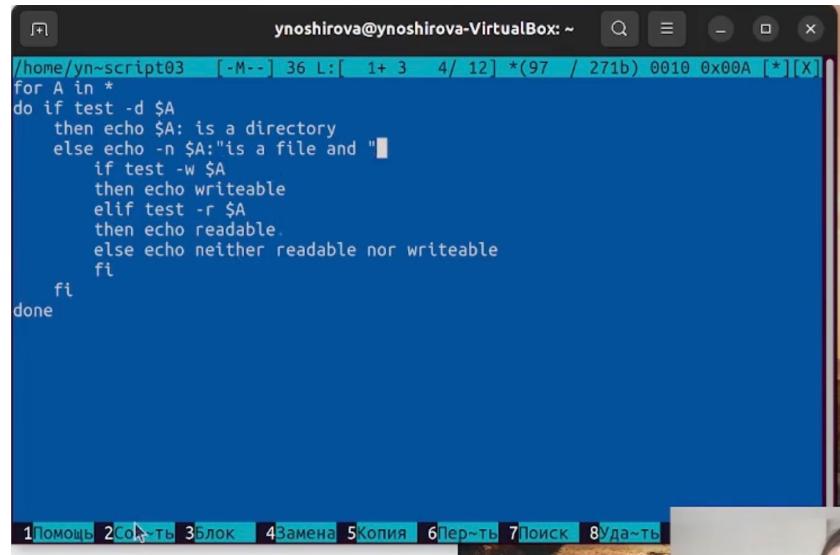
Рис. 4.3: Текст второй программы



```
ynoshirova@ynoshirova-VirtualBox:~/backup$ mcedit script01
ynoshirova@ynoshirova-VirtualBox:~/backup$ cd ..
ynoshirova@ynoshirova-VirtualBox:~$ mcedit script02
ynoshirova@ynoshirova-VirtualBox:~$ chmod +x script02
ynoshirova@ynoshirova-VirtualBox:~$ ./script02 1 2 3 4
1: 1
2: 2
3: 3
4: 4
ynoshirova@ynoshirova-VirtualBox:~$ ./script02 1 2 3 4 5 6 7 8 9 10 11
1: 1
2: 2
3: 3
4: 4
5: 5
6: 6
7: 7
8: 8
9: 9
10: 10
11: 11
ynoshirova@ynoshirova-VirtualBox:~$
```

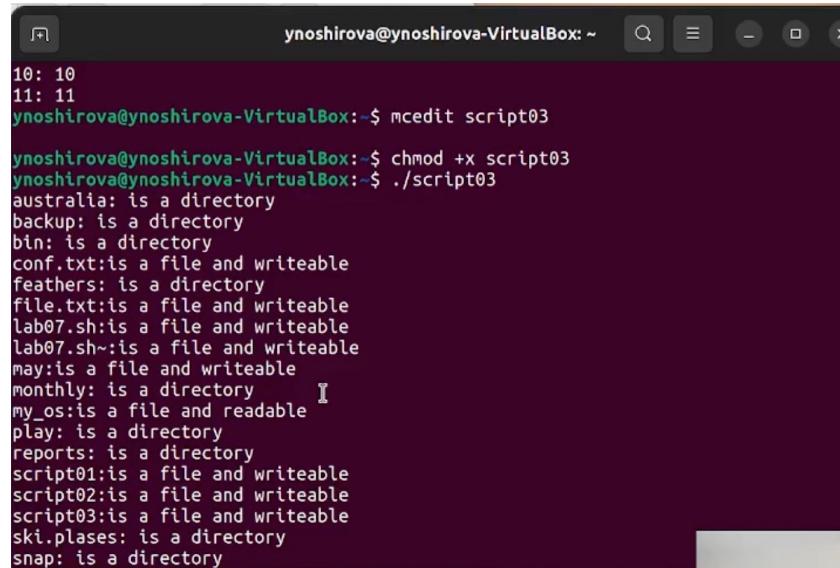
Рис. 4.4: Проверка работы второй программы

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.



```
/home/yn~script03 [-M--] 36 L:[ 1+ 3 4/ 12] *(97 / 271b) 0010 0x00A [*][X]
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
        elif test -r $A
            then echo readable.
        else echo neither readable nor writeable
        fi
    fi
done
```

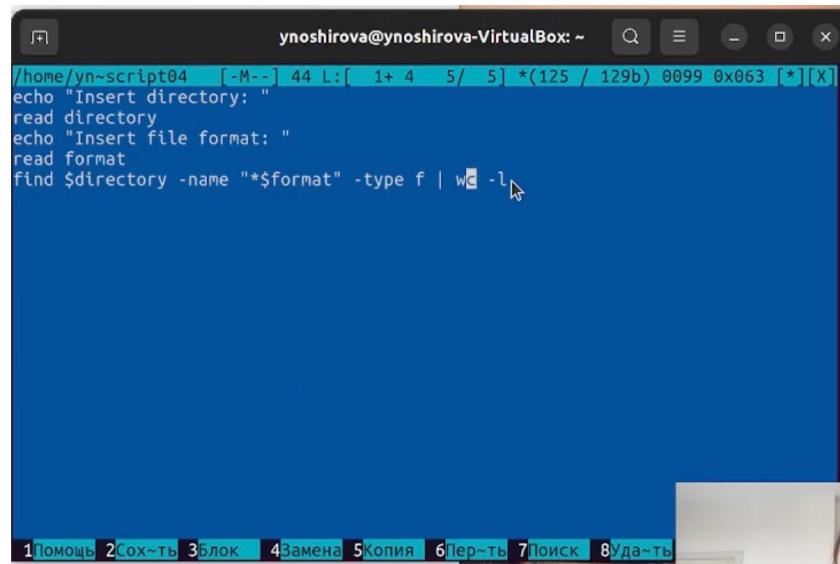
Рис. 4.5: Текст третьей программы



```
10: 10
11: 11
ynoshirova@ynoshirova-VirtualBox:~$ mcedit script03
ynoshirova@ynoshirova-VirtualBox:~$ chmod +x script03
ynoshirova@ynoshirova-VirtualBox:~$ ./script03
australia: is a directory
backup: is a directory
bin: is a directory
conf.txt:is a file and writeable
feathers: is a directory
file.txt:is a file and writeable
lab07.sh:is a file and writeable
lab07.sh~:is a file and writeable
may:is a file and writeable
monthly: is a directory      I
my_os:is a file and readable
play: is a directory
reports: is a directory
script01:is a file and writeable
script02:is a file and writeable
script03:is a file and writeable
ski.places: is a directory
snap: is a directory
```

Рис. 4.6: Проверка работы третьей программы

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.



```
/home/yn~script04 [-M--] 44 L:[ 1+ 4 5/ 5] *(125 / 129b) 0099 0x063 [*][X]
echo "Insert directory: "
read directory
echo "Insert file format: "
read format
find $directory -name "*$format" -type f | wc -l
```

Рис. 4.7: Текст четвертой программы

```
Музыка: is a directory
Общедоступные: is a directory
./script03: строка 2: test: Рабочий: ожидается бинарный оператор
Рабочий стол:is a file and ./script03: строка 5: test: Рабочий: ожидается бинарный оператор
./script03: строка 7: test: Рабочий: ожидается бинарный оператор
neither readable nor writeable
Шаблонны: is a directory
ynoshirova@ynoshirova-VirtualBox:~$ mcedit script04
ynoshirova@ynoshirova-VirtualBox:~$ chmod +x script04
ynoshirova@ynoshirova-VirtualBox:~$ ./script04
Insert directory:
/home/ynoshirova
Insert file format:
.pdf
33
ynoshirova@ynoshirova-VirtualBox:~$ ./script04
Insert directory:
/home/ynoshirova
Insert file format:
.txt
17
ynoshirova@ynoshirova-VirtualBox:~$
```

Рис. 4.8: Проверка работы четвертой программы

5 Выводы

В процессе выполнения лабораторной работы я изучила основы программирования в оболочке ОС UNIX/Linux. Научилась писать небольшие командные файлы.

Список литературы

1. Лабораторная работа № 10. Программирование в командном процессоре ОС UNIX. Командные файлы [Электронный ресурс]