

Отчёт по лабораторной работе №12

Операционные системы

Чистов Даниил Максимович

Содержание

| | | |
|----------|---------------------------------------|-----------|
| 1 | Цель работы | 4 |
| 2 | Задания | 5 |
| 3 | Выполнение лабораторной работы | 6 |
| 4 | Программа 1 | 7 |
| 5 | Программа 2 | 9 |
| 6 | Программа 3 | 11 |
| 7 | Программа 4 | 14 |
| 8 | Выводы | 16 |
| | Список литературы | 17 |

Список иллюстраций

| | | |
|-----|---------------------------------|----|
| 4.1 | Задание 1 - выполнено | 7 |
| 4.2 | Задание 1 - программа | 8 |
| 5.1 | Задание 2 - выполнено | 9 |
| 5.2 | Задание 2 - программа | 10 |
| 6.1 | Задание 3 - начало | 11 |
| 6.2 | Задание 3 - программа | 12 |
| 6.3 | Задание 3 - выполнено | 13 |
| 7.1 | Задание 4 - начало | 14 |
| 7.2 | Задание 4 - программа | 15 |
| 7.3 | Задание 4 - выполнено | 15 |

1 Цель работы

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

2 Задания

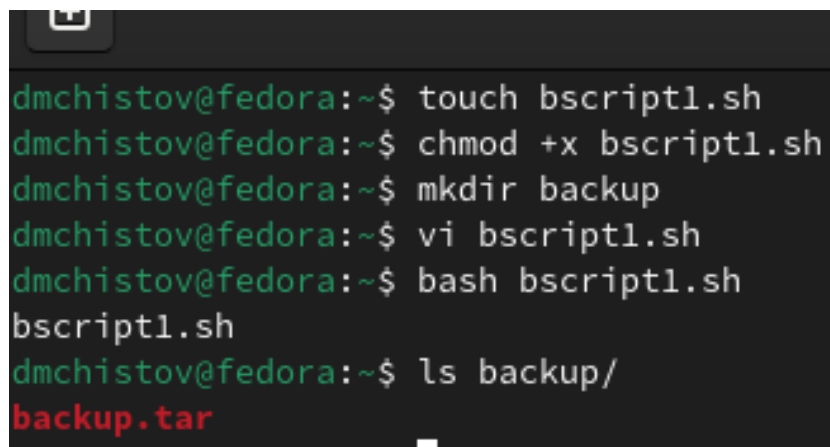
1. Программа 1
2. Программа 2
3. Программа 3
4. Программа 4

3 Выполнение лабораторной работы

4 Программа 1

Задание: “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.”

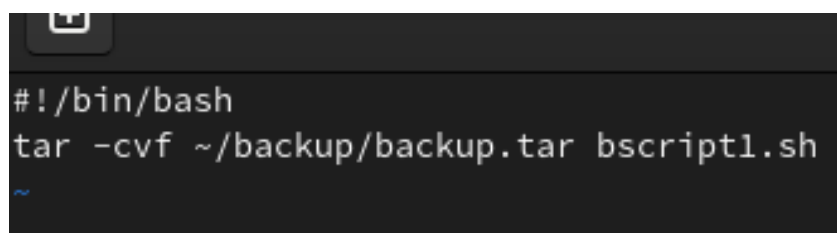
Приступаю к выполнению работы. Создаю файл с кодом, создаю папку, куда будет сохраняться архив. Пишу код (сама программа будет показана позже), выполняю - готово (рис. 4.1).

A terminal window with a dark background and green text. The user 'dmchistov' is at a 'fedora' machine in the home directory '~'. The commands and their outputs are: 'touch bscript1.sh' (no output), 'chmod +x bscript1.sh' (no output), 'mkdir backup' (no output), 'vi bscript1.sh' (no output), 'bash bscript1.sh' (output: 'bscript1.sh'), and 'ls backup/' (output: 'backup.tar').

```
dmchistov@fedora:~$ touch bscript1.sh
dmchistov@fedora:~$ chmod +x bscript1.sh
dmchistov@fedora:~$ mkdir backup
dmchistov@fedora:~$ vi bscript1.sh
dmchistov@fedora:~$ bash bscript1.sh
bscript1.sh
dmchistov@fedora:~$ ls backup/
backup.tar
```

Рис. 4.1: Задание 1 - выполнено

Код программы 1: утилитой tar задаю место, где будет архив, а затем, что нужно архивировать (рис. 4.2).

A terminal window with a dark background. The title bar shows a window icon and the text "Terminal". The content of the terminal is a shell script:

```
#!/bin/bash  
tar -cvf ~/backup/backup.tar bscript1.sh  
~  
$
```

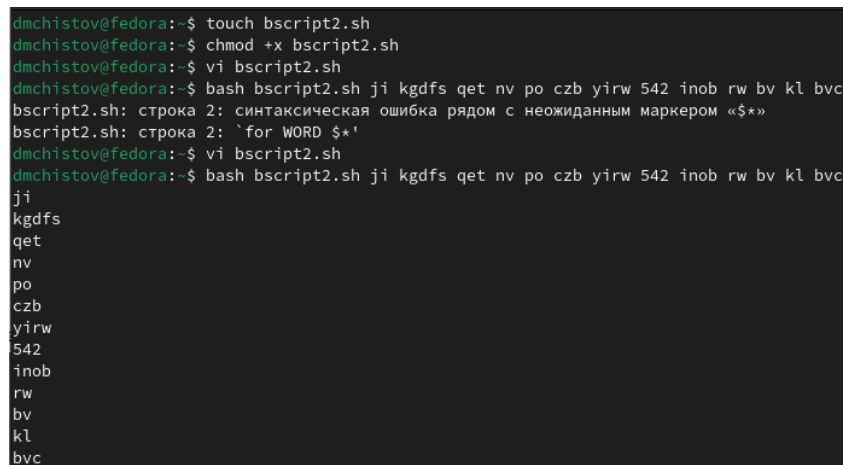
```
#!/bin/bash  
tar -cvf ~/backup/backup.tar bscript1.sh  
~  
$
```

Рис. 4.2: Задание 1 - программа

5 Программа 2

Задание: “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.”

Приступаю к выполнению - создаю, даю права, пишу код, исправляю ошибки, запускаю - работает (рис. 5.1).



```
dmchistov@fedora:~$ touch bscript2.sh
dmchistov@fedora:~$ chmod +x bscript2.sh
dmchistov@fedora:~$ vi bscript2.sh
dmchistov@fedora:~$ bash bscript2.sh ji kgdfs qet nv po czb yirw 542 inob rw bv kl bvc
bscript2.sh: строка 2: синтаксическая ошибка рядом с неожиданным маркером «$*»
bscript2.sh: строка 2: `for WORD $*'
dmchistov@fedora:~$ vi bscript2.sh
dmchistov@fedora:~$ bash bscript2.sh ji kgdfs qet nv po czb yirw 542 inob rw bv kl bvc
ji
kgdfs
qet
nv
po
czb
yirw
542
inob
rw
bv
kl
bvc
```

Рис. 5.1: Задание 2 - выполнено

Код программы 2: Циклом `for` прохожусь по каждому поданному слову в строке, которая подаётся с запуском, и каждое это слово вывожу (рис. 5.2).

```
#!/bin/bash
for WORD in $*
do echo $WORD
done
```

~

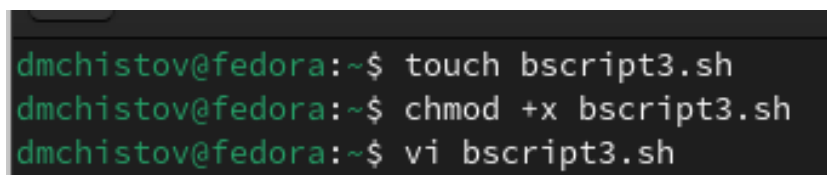
~

Рис. 5.2: Задание 2 - программа

6 Программа 3

Задание: “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.”

Создаю файл, даю права, начинаю писать код (рис. 6.1).



```
dmchistov@fedora:~$ touch bscript3.sh
dmchistov@fedora:~$ chmod +x bscript3.sh
dmchistov@fedora:~$ vi bscript3.sh
```

Рис. 6.1: Задание 3 - начало

Код программы 3: Подаю путь, утилитой `test` и её параметрами проверяю файл: выполняемый ли он, читаемый ли, доступный для записи, вывожу соответствующие строки по результатам (рис. 6.2).

```
1  #!/bin/bash
2  for X in *
3  do
4      echo "File - $X is:"
5      if test -d $X
6      then
7          echo " + a directory"
8      else
9          if test -e $X
10         then
11             echo " + is executable"
12
13             if test -w $X
14             then
15                 echo " + is writeable"
16
17                 if test -r $X
18                 then
19                     echo " + is readable"
20                 fi
21             fi
22         fi
23     fi
24 done
```

Рис. 6.2: Задание 3 - программа

Запускаю, всё работает (рис. 6.3).

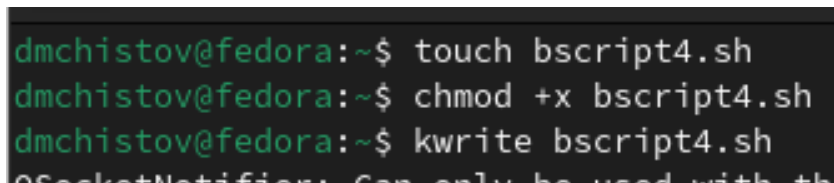
```
dmchistov@fedora:~$ bash bscript3.sh work/
File - abcl is:
+ is executable
+ is writeable
+ is readable
File - australia is:
+ a directory
File - backup is:
+ a directory
File - bscript1.sh is:
+ is executable
+ is writeable
+ is readable
File - bscript2.sh is:
+ is executable
+ is writeable
+ is readable
File - bscript3.sh is:
+ is executable
+ is writeable
+ is readable
File - conf.txt is:
+ is executable
```

Рис. 6.3: Задание 3 - выполнено

7 Программа 4

Задание: “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”

Создаю файл, даю права, начинаю писать код (рис. 7.1).



```
dmchistov@fedora:~$ touch bscript4.sh
dmchistov@fedora:~$ chmod +x bscript4.sh
dmchistov@fedora:~$ kwrite bscript4.sh
OSocketNotifier: Can only be used with th
```

Рис. 7.1: Задание 4 - начало

Код программы 4: Задаю переменные места для поиска и тип файлов. Прошу записать тип файла без точки, затем считываю. Прошу записать место поиска, затем считываю. Утилитой `find` вписываю место поиска, критерий поиска (имя), тип файла, и указываю, что нужно найти именно файлы, а не например директории. Затем использую утилиту `wc`, чтобы подсчитать количество найденных файлов (рис. 7.2).

```

1  #!bin/bash
2  path=""
3  type=""
4  echo "Enter file type (WITHOUT DOT): "
5  read type
6  echo "Enter search location (name of directory) "
7  read path
8  find "$path" -name ".*$type" -type f | wc -l
9

```

Рис. 7.2: Задание 4 - программа

Запускаю, всё работает (рис. 7.3).

```

dmchistov@fedora:~$ bash bscript4.sh
Enter file type (WITHOUT DOT):
png
Enter search location (name of directory)
work
489

```

Рис. 7.3: Задание 4 - выполнено

8 Выводы

В результате выполнения данной работы я изучил основы программирования в оболочке ОС UNIX/Linux. Научился писать небольшие командные файлы.

Список литературы

Лабораторная работы №12 Команда find