

Отчёт по лабораторной работе №11

Программирование в командном процессоре ОС UNIX. Командные файлы

Дисциплина: Операционные системы

Студент: Оразгелдиева Огулнур

Группа: НПИбд-02-20

Студ. номер: 1032205431

2021, Москва

Лабораторная работа №11

Программирование в командном процессоре ОС UNIX. Командные файлы

Цель:

- изучить основы программирования в оболочке ОС UNIX/Linux.
 - научиться писать небольшие командные файлы.
-

Задачи:

1. Ознакомиться с теоретическим материалом
 2. Написать скрипт, который при запуске будет делать резервную копию самого себя в другую директорию backup в вашем домашнем каталоге. При этом файл должен архивироваться одним из архиваторов на выбор zip, bzip2 или tar.
 3. Написать пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять.
 4. 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.
 5. 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.
-

Теоретические сведения [\[1\]](#) [\[2\]](#)

Командный процессор (командная оболочка, интерпретатор команд shell) — это программа, позволяющая пользователю взаимодействовать с операционной системой компьютера.

Bash (Bourne again shell или «возрождённый» shell) – это модифицированная версия программной оболочки Bourne-shell (sh или «Оболочка Борна»). Она является командным процессором, работающим интерактивно в текстовом окне. Bash нужен для приема команд пользователя и их отправки операционной системе для последующей обработки.

Взаимодействие оболочки и операционной системы обеспечивается с помощью специальной программы – терминала.

Командный процессор bash обеспечивает возможность использования переменных типа строка символов. Имена переменных могут быть выбраны пользователем. Пользователь имеет возможность присвоить переменной значение некоторой строки символов.

Использование значения, присвоенного некоторой переменной, называется подстановкой. Для того чтобы имя переменной не сливалось с символами, которые могут следовать за ним в командной строке, при подстановке в общем случае используется следующая форма записи: \${имя переменной}

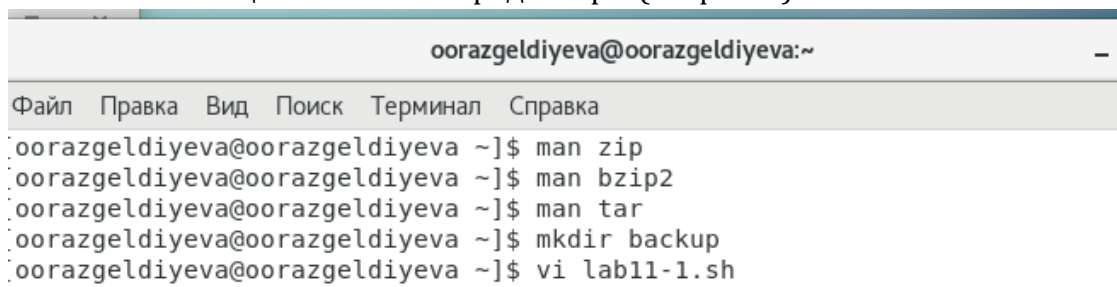
Оболочка bash позволяет работать с массивами. Для создания массива используется команда set с флагом -A.

Оболочка bash поддерживает встроенные арифметические функции. Команда let является показателем того, что последующие аргументы представляют собой выражение, подлежащее вычислению.

Последовательность команд может быть помещена в текстовый файл. Такой файл называется командным. Далее этот файл можно выполнить по команде: *bash командный_файл [аргументы]*

Ход работы

1. Создаём каталог *backup* с помощью команды *mkdir*. (см. рис. 1) Создаём файл *lab11-1.sh* с помощью текстового редактора. (см. рис. 1)



```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
oorazgeldiyeva@oorazgeldiyeva ~]$ man zip  
oorazgeldiyeva@oorazgeldiyeva ~]$ man bzip2  
oorazgeldiyeva@oorazgeldiyeva ~]$ man tar  
oorazgeldiyeva@oorazgeldiyeva ~]$ mkdir backup  
oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-1.sh
```

Рисунок 1. Создание каталога *backup* и файла *lab11-1.sh*

После открытия файла, пишем программу, по которому копируется командный файл (*lab11-2.sh*) в каталог *backup* и архивируется с помощью архиватора *bzip2*. (см. рис. 2)

```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
#!/bin/bash  
name="$0"  
cp ${name} ~/backup  
cd ~/backup  
bzip2 -k ${name}  
  
~  
~  
~  
~
```

Рисунок 2. Программа к заданию 1

Выполняем программу с помощью *bash*. (см. рис. 3)

Пояснения: задаем переменную *name* с названием данного командного файла (с указателем \$0); далее копируем командный файл в каталог *~/backup* с помощью команды *cp*. Переходим в этот каталог и архивируем файл в *bz2* и при этом сохраняем сам файл. В итоге внутри каталога получаем два файла: командный файл *lab11-1.sh* и архив с этим командным файлом *lab11-1.sh.bz2*. (см. рис 3-4)

```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
[oorazgeldiyeva@oorazgeldiyeva ~]$ man zip  
[oorazgeldiyeva@oorazgeldiyeva ~]$ man bzip2  
[oorazgeldiyeva@oorazgeldiyeva ~]$ man tar  
[oorazgeldiyeva@oorazgeldiyeva ~]$ mkdir backup  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-1.sh  
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-1.sh  
[oorazgeldiyeva@oorazgeldiyeva ~]$ ls  
004-lab_proc.pdf  lab08  lab11-1.sh  work  Изображения  Шаблоны  
backup            #lab10.sh#  #lab.txt#  Видео  Музыка  
c.cpp            lab10.sh  lab.txt  Документы  Общедоступные  
file.txt         lab10.sh~  text.txt  Загрузки  Рабочий стол  
[oorazgeldiyeva@oorazgeldiyeva ~]$ ls backup  
lab11-1.sh  lab11-1.sh.bz2
```

Рисунок 3. Содержимого домашнего каталога и каталога *backup*

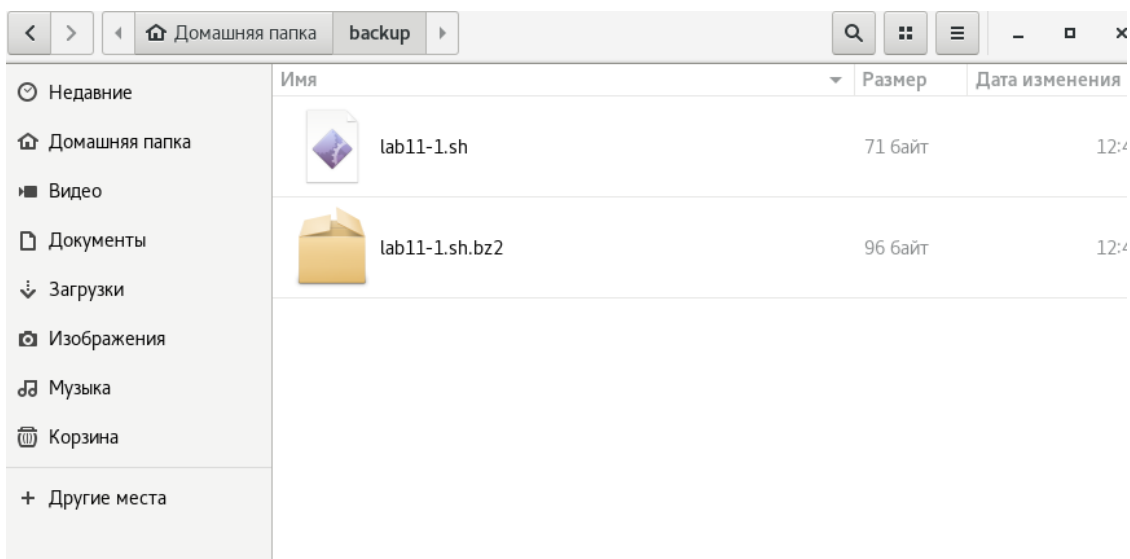


Рисунок 4. Содержимого каталога backup

Откроем полученный архив и посмотрим его содержимое. (см. рис. 5)

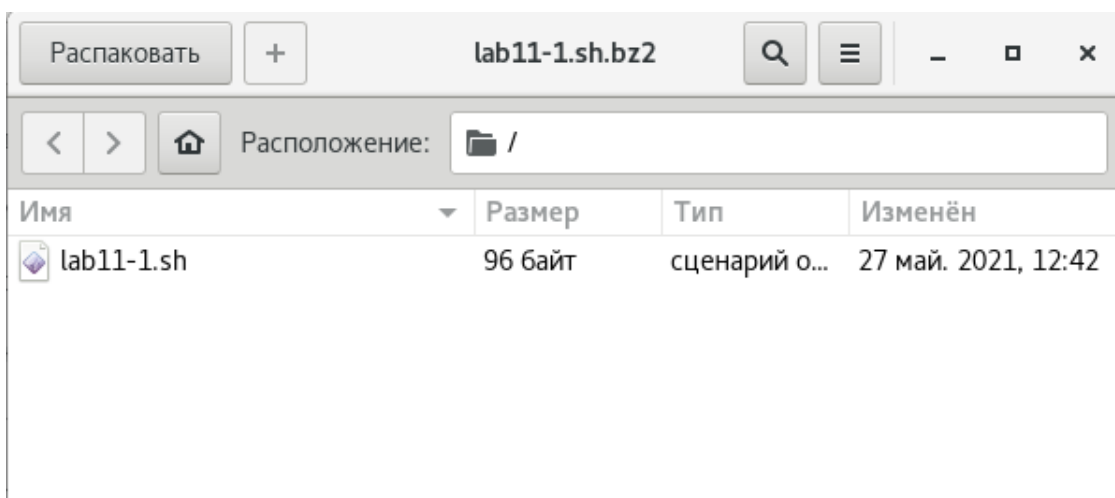


Рисунок 5. Архив

- Создадим и откроем файл *lab11-2.sh* с программой, обрабатывающей любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов. (см. рис. 6)

```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
#!/bin/bash  
set -a arguments  
read -a arguments  
echo "Result:"  
echo ${arguments[@]}  
~
```

Рисунок 6. Программа к заданию 2

Пояснения: сначала объявляем массив для аргументов с помощью команды *set* и флага *-a*; читаем вводимые аргументы с помощью *read* и выводим их на экран с помощью *echo*.

Выполняем командный файл при помощи *bash* и вводим произвольные аргументы. (см. рис. 7)

```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-2.sh  
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-2.sh  
jhsnc 9 2 9 utw o a j g p u  
Result:  
jhsnc 9 2 9 utw o a j g p u  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-2.sh
```

Рисунок 7. Выполнение программы

3. Создадим и откроем командный файл *lab11-3.sh* при помощи текстового редактора *vi* и напишем программу. (см. рис. 8)

```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-2.sh  
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-2.sh  
jhsnc 9 2 9 utw o a j g p u  
Result:  
jhsnc 9 2 9 utw o a j g p u  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-3.sh
```

Рисунок 8. Создание файла при помощи *vi*

Программа - 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. (см. рис. 9)

```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл Правка Вид Поиск Терминал Справка  
#!/bin/bash  
echo "Enter full name of directory"  
read name  
cd ${name}  
stat -c "%A %n" *  
~  
~
```

Рисунок 9. Программа к заданию 3

Пояснения: выводим запрос на ввод полного имени каталога; чтение названия каталога и переход в него; выводим на экран права доступа к файлам (при помощи %A) и название файлов (%n), использу команду *stat* с опцией *-c*. (в конце указываем "*", что означает "текущий каталог") [3]

Выполняем программу с помощью *bash*. Вводим название директории, и в результате выводятся файлы с правами доступа к ним. (см. рис. 10) (в первом случае была ошибка, так как не указала * для определения текущего каталога)

```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл Правка Вид Поиск Терминал Справка  
Result:  
jhsnc 9 2 9 utw o a j g p u  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-3.sh  
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-3.sh  
Enter full name of directory  
backup  
stat: не удалось выполнить stat для «%A %n»: Нет такого файла или каталога  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-3.sh  
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-3.sh  
Enter full name of directory  
backup  
-rw-rw-r-- lab11-1.sh  
-rw-rw-r-- lab11-1.sh.bz2
```

Рисунок 10. Выполнение программы

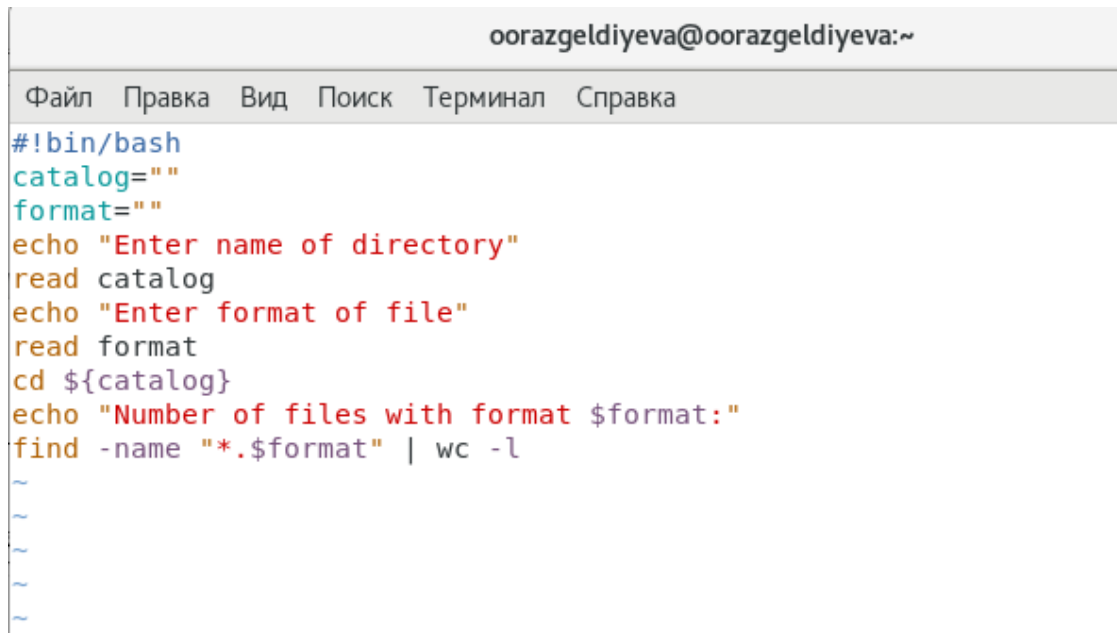
Как видно по рис. 10, для примера ввела название ранее созданного каталога *backup* (задание 1) и проверила файлы и права доступа к этим файлам, находящимся внутри этого каталога.

4. Создадим и откроем командный файл *lab11-4.sh* при помощи текстового редактора *vi* и напишем программу. (см. рис. 11)

```
backup  
-rw-rw-r-- lab11-1.sh  
-rw-rw-r-- lab11-1.sh.bz2  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-4.sh  
~  
~
```

Рисунок 11. Создание файла

Напишем программу, 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. (см. рис. 12)



```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
#!/bin/bash  
catalog=""  
format=""  
echo "Enter name of directory"  
read catalog  
echo "Enter format of file"  
read format  
cd ${catalog}  
echo "Number of files with format $format:"  
find -name ".*$format" | wc -l  
~  
~  
~  
~  
~
```

Рисунок 12. Программа к заданию 4

Пояснения: создаем переменные для названия каталога и формата файла (*catalog* и *format* соответственно); запрос на ввод названия каталога с помощью *echo* и его чтение; аналогично вводим формат. Переходим в заданный каталог и с помощью команды *find*, конвейера (*|*) и команды *wc -l* находим (*find*) и выводим количество файлов (*wc -l*) с указанным форматом.

Выполняем программу с помощью *bash*. Вводим название каталога (*backup*) и формат *bz2*. (см. рис. 13) (в первом случае сначала вводили формат, а потом выводился запрос на формат, поэтому исправила программу и выполнила еще раз)

```
oorazgeldiyeva@oorazgeldiyeva:~  
Файл Правка Вид Поиск Терминал Справка  
backup  
-rw-rw-r-- lab11-1.sh  
-rw-rw-r-- lab11-1.sh.bz2  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-4.sh  
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-4.sh  
Enter name of directory  
backup  
bz2  
Enter format of file  
Number of files with format bz2:  
1  
[oorazgeldiyeva@oorazgeldiyeva ~]$ vi lab11-4.sh  
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-4.sh  
Enter name of directory  
backup  
Enter format of file  
bz2  
Number of files with format bz2:  
1
```

Рисунок 13. Выполнение программы

Также для примера посмотрим, сколько файлов с расширением *.txt* в каталоге *work*, с расширением *.sh* в *backup*. (см. рис. 14)

```
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-4.sh  
Enter name of directory  
work  
Enter format of file  
txt  
Number of files with format txt:  
1  
[oorazgeldiyeva@oorazgeldiyeva ~]$ bash lab11-4.sh  
Enter name of directory  
backup  
Enter format of file  
sh  
Number of files with format sh:  
1  
[oorazgeldiyeva@oorazgeldiyeva ~]$
```

Рисунок 14. Выполнение программы

Вывод: на лабораторной работе изучила основы программирования в оболочке ОС UNIX/Linux; научилась писать небольшие командные файлы.

Библиография [1] - РУДН, Операционные системы, Программирование в командном процессоре ОС UNIX. Командные файлы

[2] - Linux. Bash. Что это? Основы bash

[3] - Команда stat в Linux