**Липецкий государственный технический университет**

Факультет автоматизации и информатики

Кафедра автоматизированных систем управления

ЛАБОРАТОРНАЯ РАБОТА №5

по дисциплине «ДПО Интаро»

Программирование в ОС семейства Linux

Программирование на SHELL. Использование командных файлов

|  |  |  |
| --- | --- | --- |
| Студент |  | Поленников А.А. |
| Группа АИ-17 |  | . |
| Руководитель |  |  |
|  |  | Кургасов В.В. |

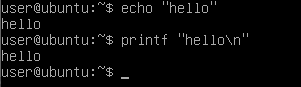
Липецк 2019 г.

Цель работы

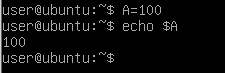
Изучение основных возможностей языка программирования Shell с целью автоматизации процесса администрирования системы за счет написания и использования командных файлов.

Ход выполнения работы

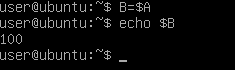
1. Используя команды ECHO, PRINTF вывести информационные сообщения на экран.



2. Присвоить переменной А целочисленное значение. Просмотреть значение переменной А.



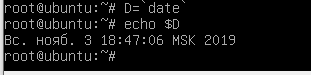
3. Присвоить переменной В значение переменной А. Просмотреть значение переменной В.



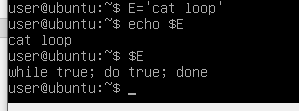
4. Присвоить переменной С значение “путь до своего каталога”. Перейти в этот каталог с использованием переменной.



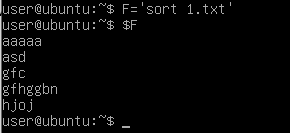
5. Присвоить переменной D значение “имя команды”, а именно, команды DATE. Выполнить эту команду, используя значение переменной.



6. Присвоить переменной E значение “имя команды”, а именно, команды просмотра содержимого файла, просмотреть содержимое переменной. Выполнить эту команду, используя значение переменной.



7. Присвоить переменной F значение “имя команды”, а именно сортировки содержимого текстового файла. Выполнить эту команду, используя значение переменной.



Написать скрипты, при запуске которых выполняются следующие действия:

8. Программа запрашивает значение переменной, а затем выводит значение этой переменной.



Код скрипта

echo “please enter”

read a

echo $a

9. Программа запрашивает имя пользователя, затем здоровается с ним, используя значение введенной переменной.



Код скрипта

echo “enter a name”

read name

echo “Hello “$name

10. Программа запрашивает значения двух переменных, вычисляет сумму (разность, произведение, деление) этих переменных. Результат выводится на экран (использовать команды a) EXPR; б) ВС).,



Код скрипта

echo “enter a”

rad a

echo “enter b”

read b

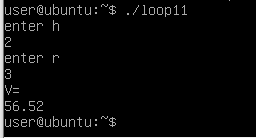
echo “EXPR “

expr $a + $b

echo “BC “

echo “$a - $b” | bc

11. Вычислить объем цилиндра. Исходные данные запрашиваются программой. Результат выводится на экран.



Код скрипта

Pi =3.14

Echo “enter h”

Read h

Echo “enter r”

Read r

Echo “V=”

Echo “$pi \* $r \*$r \*$h” |bc

12. Используя позиционные параметры, отобразить имя программы, количество аргументов командной строки, значение каждого аргумента командной строки.



Код скрипта

Echo $0

Echo $#

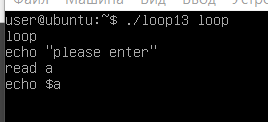
For x in $@

do

Echo $x

Done

13. Используя позиционный параметр, отобразить содержимое текстового файла, указанного в качестве аргумента командной строки. После паузы экран очищается.



Код скрипта

Echo $1

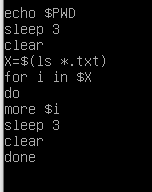
Cat $1

Sleep 3s

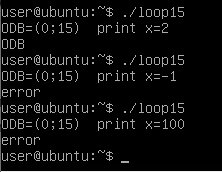
Clear

14. Используя оператор FOR, отобразить содержимое текстовых файлов текущего каталога поэкранно.

Код скрипта



15. Программой запрашивается ввод числа, значение которого затем сравнивается с допустимым значением. В результате этого сравнения на экран выдаются соответствующие сообщения.



Код скрипта

Echo -n “ODB=(0;15) print x=”

Read x

If[[ $x -gt 0 && $x -lt 15]]

Then

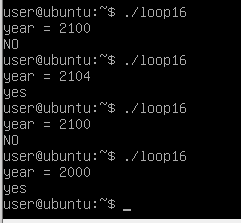
Echo “OBD”

Else

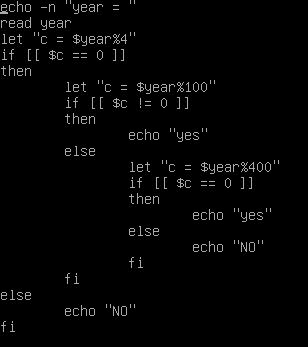
Echo “error”

fi

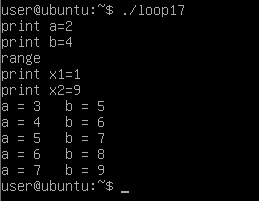
16. Программой запрашивается год, определяется, високосный ли он. Результат выдается на экран.



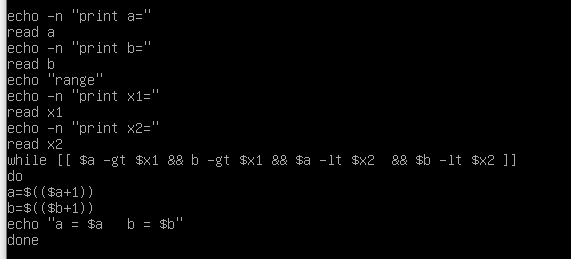
Код скрипта



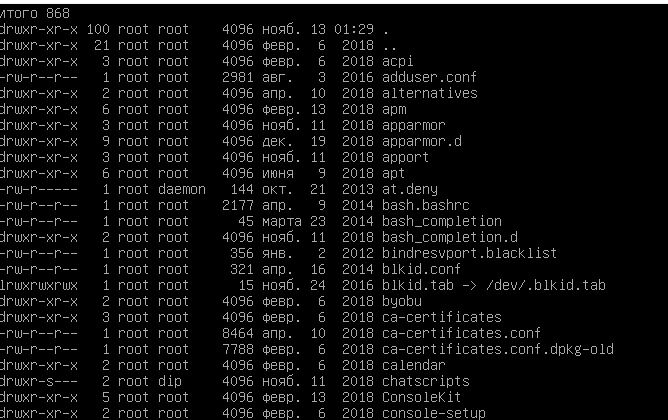
17. Вводятся целочисленные значения двух переменных. Вводится диапазон данных. Пока значения переменных находятся в указанном диапазоне, их значения инкрементируются.



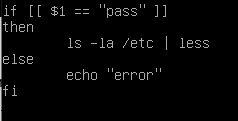
Код скрипта



18. В качестве аргумента командной строки указывается пароль. Если пароль введен верно, постранично отображается в длинном формате с указанием скрытых файлов содержимое каталога /etc.



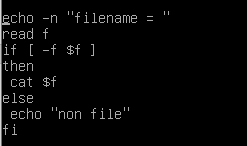
Код скрипта



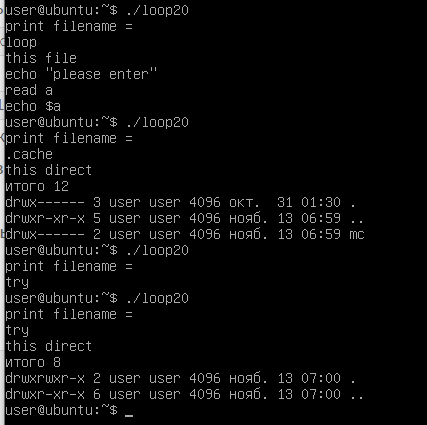
19. Проверить, существует ли файл. Если да, выводится на экран его содержимое, если нет - выдается соответствующее сообщение.



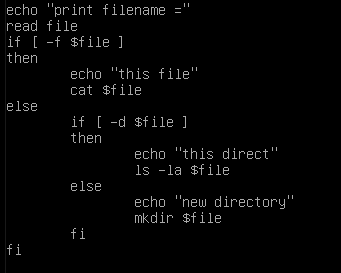
Код скрипта



20. Если файл есть каталог и этот каталог можно читать, просматривается содержимое этого каталога. Если каталог отсутствует, он создается. Если файл не есть каталог, просматривается содержимое файла.

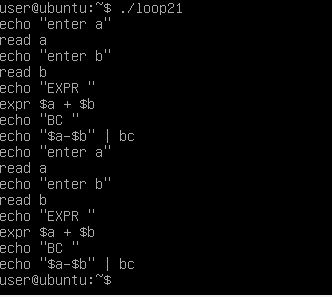


Код скрипта

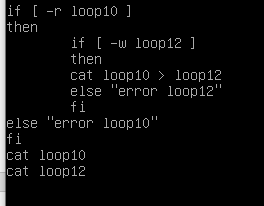


21. Анализируются атрибуты файла. Если первый файл существует и используется для чтения, а второй файл существует и используется для записи, то содержимое первого файла перенаправляется во второй файл. В случае несовпадений указанных атрибутов или отсутствия файлов на экран выдаются соответствующие сообщения (использовать а) имена файлов; б) позиционные параметры).

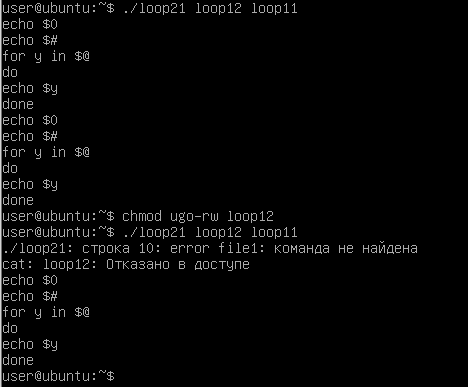
а) имена файлов



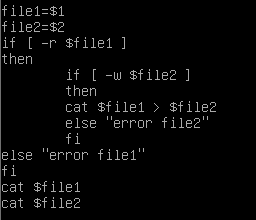
Код скрипта



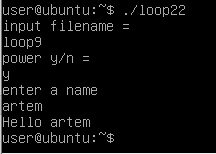
б) позиционные параметры



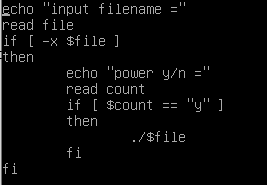
Код скрипта



22. Если файл запуска программы найден, программа запускается (повыбору).



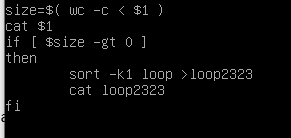
Код скрипта



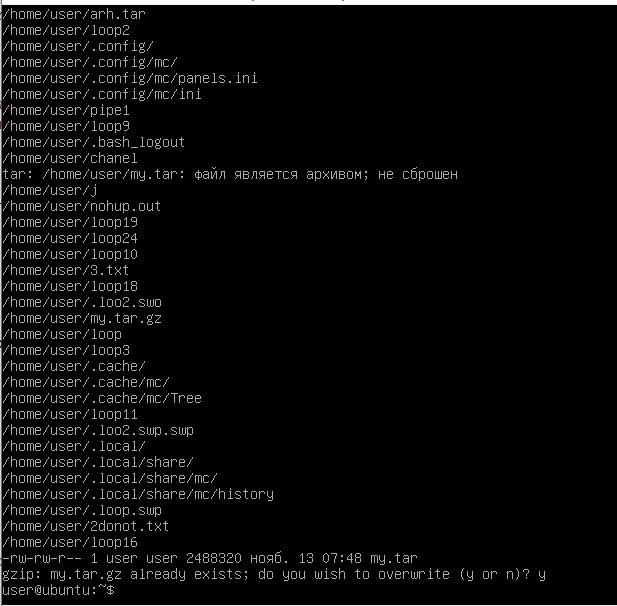
23. В качестве позиционного параметра задается файл, анализируется его размер. Если размер файла больше нуля, содержимое файла сортируется по первому столбцу по возрастанию, отсортированная информация помещается в другой файл, содержимое которого затем отображается на экране.



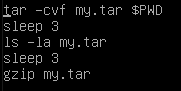
Код скрипта



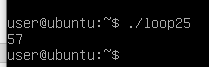
24. Командой TAR осуществляется сборка всех текстовых файлов текущего каталога в один архивный файл my.tar, после паузы просматривается содержимое файла my.tar, затем командой GZIP архивный файл my.tar сжимается.



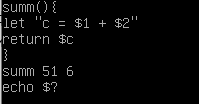
Код скрипта



25. Написать скрипт с использованием функции, например, функции, суммирующей значения двух переменных.



Код скрипта



Вывод

В ходе выполнения данной работы были изучены основные возможности языка программирования Shell с целью автоматизации процесса администрирования системы за счет написания и использования командных файлов.