

Лабораторная работа №11

Колосова Кристина Александровна

Цель работы

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

Теоретическая часть

Пользователь имеет возможность присвоить переменной значение некоторой строки символов. Значение, присвоенное некоторой переменной, может быть впоследствии использовано.

Такие символы, как ' < > * ? | \ " &, являются метасимволами и имеют для командного процессора специальный смысл. Снятие специального смысла с метасимвола называется экранированием метасимвола. Экранирование может быть осуществлено с помощью предшествующего метасимволу символа \, который, в свою очередь, является метасимволом.

Последовательность команд может быть помещена в текстовый файл. Такой файл называется командным. Далее этот файл можно выполнить по команде:

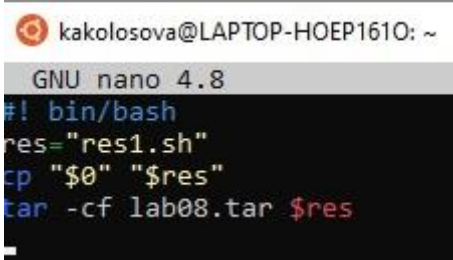
```
bash командный_файл [аргументы]
```

Чтобы не вводить каждый раз последовательности символов bash, необходимо изменить код защиты этого командного файла, обеспечив доступ к этому файлу по выполнению. Это может быть сделано с помощью команды `chmod +x имя_файла`

При написании команд используется C подобный синтаксис. Благодаря этому можно писать циклы привычным способом.

Выполнение лабораторной работы

1. Написала скрипт (рис. - @fig:001), который при запуске будет делать резервную копию самого себя. При этом файл архивируется архиватором tar. (рис. - @fig:002)



The screenshot shows a terminal window with the prompt `kakolosova@LAPTOP-HOEP161O: ~`. Below it, the GNU nano 4.8 editor is open, displaying a shell script. The script's content is as follows:

```
#!/ bin/bash
res="res1.sh"
cp "$0" "$res"
tar -cf lab08.tar $res
```

Сам скрипт 1

```

kakolosova@LAPTOP-HOEP1610:/$ sudo cp ~/ress.sh res.sh
kakolosova@LAPTOP-HOEP1610:/$ ls
bin  dev  home  lib  lib64  lost+found  mnt  proc  root  sbin  srv  tmp  var
boot  etc  init  lib32  libx32  media  opt  res.sh  run  snap  sys  usr
kakolosova@LAPTOP-HOEP1610:/$ sudo chmod +x res.sh
kakolosova@LAPTOP-HOEP1610:/$ nano res.sh
kakolosova@LAPTOP-HOEP1610:/$ ./res.sh
cp: cannot create regular file 'res1.sh': Permission denied
tar: lab08.tar: Cannot open: Permission denied
tar: Error is not recoverable: exiting now
kakolosova@LAPTOP-HOEP1610:/$ sudo ./res.sh
kakolosova@LAPTOP-HOEP1610:/$ ls
bin  dev  home  lab08.tar  lib32  libx32  media  opt  res.sh  root  sbin  srv  tmp  var
boot  etc  init  lib  lib64  lost+found  mnt  proc  res1.sh  run  snap  sys  usr

```

Проверка работы

Способ использования команд архивации узнала, изучив справку. (рис. - @fig:003)

```

TAR(1)                                GNU TAR Manual

NAME
    tar - an archiving utility

SYNOPSIS
    Traditional usage
        tar {A|c|d|r|t|u|x}[GnSkUWmpsMBiajJzZhPlRvwo] [ARG...]

    UNIX-style usage
        tar -A [OPTIONS] ARCHIVE ARCHIVE

        tar -c [-f ARCHIVE] [OPTIONS] [FILE...]

        tar -d [-f ARCHIVE] [OPTIONS] [FILE...]

        tar -t [-f ARCHIVE] [OPTIONS] [MEMBER...]

        tar -r [-f ARCHIVE] [OPTIONS] [FILE...]

        tar -u [-f ARCHIVE] [OPTIONS] [FILE...]

        tar -x [-f ARCHIVE] [OPTIONS] [MEMBER...]

    GNU-style usage
        tar {--catenate|--concatenate} [OPTIONS] ARCHIVE ARCHIVE

        tar --create [--file ARCHIVE] [OPTIONS] [FILE...]

```

Справка по команде

- Написала пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. (рис. - @fig:004)

```

GNU nano 4.8                                res2.sh
for i in "$*"
do
esho $i
done

```

Скрипт 2

Скрипт последовательно распечатывает значения всех переданных аргументов. (рис. - @fig:005)

```

kakolosova@LAPTOP-HOEP1610:/$ ./res2.sh 5 7 15 g y
5 7 15 g y

```

Исполнение

- Написала командный файл — аналог команды ls (без использования самой этой команды и команды dir). Он выдает информацию о нужном каталоге и выводит информацию о возможностях доступа к файлам этого каталога. (рис. - @fig:006) (рис. - @fig:007)

```

GNU nano 4.8                                ress.sh
for i in *
do if test -d $i
then echo $i: ' Dir '
else echo -n $i: ' File '
if test -w $i
then echo " to Write_"
if test -r $i
then echo " to Read"
else echo " not ready.."
fi
fi
fi
done

```

Скрипт 3

```

kakolosova@LAPTOP-HOEP1610:~$ ./ress.sh
conf.txt: File to Write_
to Read
file.txt: File to Write_
to Read
laboratory: Dir
ress.sh: File to Write_
to Read
text.txt: File to Write_
to Read
work: Dir

```

Исполнение 3 скрипта

4. Написала 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.

```

GNU nano 4.8
find $1 -name ".*$2" -type f | wc -l

```

Скрипт 4

```

kakolosova@LAPTOP-HOEP1610:~$ ls
conf.txt file.txt laboratory ress.sh text.txt work
kakolosova@LAPTOP-HOEP1610:~$ ./ress.sh /home txt
7
kakolosova@LAPTOP-HOEP1610:~$ find /home -name "*.sh" -type f
/home/kakolosova/work/os/lab09/hello.sh
/home/kakolosova/ress.sh
kakolosova@LAPTOP-HOEP1610:~$ find /home -name "*.txt" -type f
/home/kakolosova/text.txt
/home/kakolosova/file.txt
/home/kakolosova/laboratory/2020-2021/OS/.txt
/home/kakolosova/laboratory/2020-2021/OS/lab06/tyu.txt
/home/kakolosova/laboratory/2020-2021/OS/lab05/tyu.txt
/home/kakolosova/laboratory/2020-2021/OS/hello.txt
/home/kakolosova/conf.txt
kakolosova@LAPTOP-HOEP1610:~$ ./ress.sh /home sh
2

```

Запуск и проверка результата

Выводы

Изучила основы программирования в оболочке ОС UNIX/Linux. Научилась писать небольшие командные файлы.