

Министерство образования Республики Беларусь

Учреждение образования

Брестский государственный технический университет

Лабораторная работа 7

Выполнил:

Студент 3-го курса

Группы АС-51

Потапов В.Д.

Проверила:

Давидюк Ю.И.

Брест 2020

Задание 1: Вывести любое сообщение с помощью команды echo перенаправив вывод:

- в несуществующий файл с помощью символа >;
- в несуществующий файл с помощью символа >>;
- в существующий файл с помощью символа >;
- в существующий файл с помощью символа >>;

```
[MBP-Vladimir:SPO vladimirpotapov$ echo something new > second_one.txt
[MBP-Vladimir:SPO vladimirpotapov$ echo something other >> third_one.txt
[MBP-Vladimir:SPO vladimirpotapov$ cat second_one.txt
something new
[MBP-Vladimir:SPO vladimirpotapov$ cat third_one.txt
something other
[MBP-Vladimir:SPO vladimirpotapov$ echo already exists > second_one.txt
[MBP-Vladimir:SPO vladimirpotapov$ echo already exists >> third_one.txt
[MBP-Vladimir:SPO vladimirpotapov$ cat second_one.txt
already exists
[MBP-Vladimir:SPO vladimirpotapov$ cat third_one.txt
something other
already exists
```

Задание 2: Переадресовать стандартный ввод для команды cat на файл

```
[MBP-Vladimir:SPO vladimirpotapov$ cat < third_one.txt
something other
already exists
```

Задание 3: Вывести сообщение с помощью команды echo в канал ошибок. Создать файл myscript:

```
#!/bin/sh
echo stdout
echo stderr>&2
exit 0
```

- Запустить его:
 - Без перенаправления (sh myscript)

```
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh
stdout
stderr
```

- перенаправив стандартный вывод в файл, просмотреть содержимое файла (sh myscript > file1);

```
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh > first_one.txt
stderr
[MBP-Vladimir:SPO vladimirpotapov$ cat first_one.txt
stdout
```

- перенаправить стандартный канал ошибок в существующий и несуществующий файлы с помощью символов `>` и `>>`.

```
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh 2> second_one.txt
stdout
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh 2>> third_one.txt
stdout
[MBP-Vladimir:SPO vladimirpotapov$ cat second_one.txt
stderr
[MBP-Vladimir:SPO vladimirpotapov$ cat third_one.txt
something other
already exists
stderr
```

- перенаправив стандартный вывод в файл 1, стандартный канал ошибок - в файл 2.

```
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh 1> second_one.txt
stderr
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh 2> third_one.txt
stdout
[MBP-Vladimir:SPO vladimirpotapov$ cat second_one.txt
stdout
[MBP-Vladimir:SPO vladimirpotapov$ cat third_one.txt
stderr
```

- перенаправив стандартный вывод и стандартный канал ошибок в файл 3

```
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh 2> third_one.txt
stdout
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh 1>> third_one.txt
stderr
[MBP-Vladimir:SPO vladimirpotapov$ cat third_one.txt
stderr
stdout
```

- перенаправив стандартный вывод в файл 4 с помощью символа `>`, а стандартный канал ошибок в файл 4 с помощью символа `>>`.

```
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh 1> fourth_one.txt
stderr
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript.sh 2>> fourth_one.txt
stdout
[MBP-Vladimir:SPO vladimirpotapov$ cat fourth_one.txt
stdout
stderr
```

Задание 4: Вывести третью строку из последних десяти строк отсортированного в обратном порядке файла /etc/group.

Комментарий: стандартно tail делает выборку 10 строк с хвоста

```
[MBP-Vladimir:SPO vladimirpotapov$ sort -r /etc/group | tail | head -3 | tail -1  
# in single-user mode. At other times this information is provided by
```

Задание 5: Подсчитать при помощи конвейера команд количество блочных устройств ввода-вывода, доступных в системе.

Комментарий: grep ^b необходим для выбора блочных устройств из списка.

```
[MBP-Vladimir:dev vladimirpotapov$ ls -l ./ | grep '^b' | wc -l  
9
```

Задание 6: Написать скрипт, выводящий на консоль все аргументы командной строки, переданные данному скрипту. Привести различные варианты запуска данного скрипта, в том числе без непосредственного вызова интерпретатора в командной строке.

```
#!/bin/sh  
#while was reading about this lab found documentation for shift  
func. and there found structure of this code. I made some  
changes and here it is.  
while [ -n "$1" ]  
do  
echo 1  
shift  
done
```

```
[MBP-Vladimir:SPO vladimirpotapov$ sh myscript_6.sh par1 par3 par3 par4  
par1  
par3  
par3  
par4
```

Задание 7: Реализовать командный файл, позволяющий в цикле посыпать всем активным пользователям (исключая пользователя, запустившего данный командный файл) сообщение – сообщение вводится с клавиатуры. Командный файл при старте выводит имя компьютера, имя запустившего командный файл пользователя, тип операционной системы, список загруженных модулей.

```
#!/bin/sh  
  
PC=$(hostname)  
User=$(whoami)  
Sys=$(uname)
```

```

echo "PC name:$PC"
echo "UserName:$UserName"
echo "OS type:$OS"
#sorry no modules :(
for i in $(who | awk '{print $1}') #extracting first field with
awk from users info
do
echo "Channel opened (can be closed by control+D) for $i:"
write $i #messaging selected to user
echo "\nChannel closed with $i."
done

```

The image shows two side-by-side terminal windows. Both windows have a dark background and light-colored text. The left window shows the command being run: `sh ./SSP/sss.sh`. It then displays the variables: `PC name:MBP-Vladimir.lan`, `UserName:vladimirpotapov`, and `OS type:Darwin`. It continues with a message about modules, followed by a channel opening and closing sequence. The right window shows the output of the script being received by another user. It includes messages like "Message from vladimirpotapov@MBP-Vladimir.lan on ttys000 at 12:03 ...", "Using macOS so there's no modules command", and "EOF". There is also a message "Oh hi mark" and another EOF marker.

ВЫВОД: научился работе со стандартным каналом ошибок, и bash-криптами.