

Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский государственный технический университет»  
Кафедра ИИТ

**Лаботарорная работа №3**  
**«Bash. Потоки данных. Программирование»**  
По дисциплине «СПО»  
за пятый семестр

**Выполнил:**  
Студент 3 курса  
Факультета ЭИС  
Группы АС-50  
Бойченко А. Д.  
**Проверила:**  
Давидюк Ю.И.

**Брест 2020 г.**

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

- в несуществующий файл с помощью символа >;

```
ubuntu@ubuntu:~$ echo Hello > file1
```

```
ubuntu@ubuntu:~$ ls
```

```
3lab.docx Documents examples.desktop Music Public Videos
```

```
Desktop Downloads file1 Pictures Templates лаба.odt
```

```
ubuntu@ubuntu:~$ cat file1
```

```
Hello
```

- в несуществующий файл с помощью символа >>;

```
ubuntu@ubuntu:~$ echo Hello_World >> file2
```

```
ubuntu@ubuntu:~$ ls
```

```
3lab.docx Documents examples.desktop file2 Pictures Templates лаба.odt
```

```
Desktop Downloads file1 Music Public Videos
```

```
ubuntu@ubuntu:~$ cat file2
```

```
Hello_World
```

- в существующий файл с помощью символа >;

```
ubuntu@ubuntu:~$ echo Hello_World > file1
```

```
ubuntu@ubuntu:~$ cat file1
```

```
Hello_World
```

- в существующий файл с помощью символа >>;

```
ubuntu@ubuntu:~$ echo Hello >> file2
```

```
ubuntu@ubuntu:~$ cat file2
```

```
Hello_World
```

```
Hello
```

Объяснить результаты.

> - пересоздаёт и записывает.

>> - дописывает.

2. Переадресовать стандартный ввод для команды cat на файл.

```
ubuntu@ubuntu:~$ cat < file1
```

```
Hello_World
```

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);

```
ubuntu@ubuntu:~$ sh myscript
```

```
stdout
```

```
stderr
```

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

```
ubuntu@ubuntu:~$ sh myscript > file1
```

```
stderr
```

```
ubuntu@ubuntu:~$ head file1
```

```
stdout
```

- перенаправить стандартный канал ошибок в существующий и несуществующий файлы с помощью символов > и >> (а тут и дальше уже сами :) );

```
ubuntu@ubuntu:~$ sh myscript 2> file1
```

```
stdout
```

```
ubuntu@ubuntu:~$ cat file1
```

```
stderr
```

```
ubuntu@ubuntu:~$ sh myscript 2>> file1
```

```
stdout
```

```
ubuntu@ubuntu:~$ cat file1
```

```
stderr
```

```
stderr
```

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

```
ubuntu@ubuntu:~$ sh myscript > file1 && sh myscript 2> file2
```

```
stderr
```

```
stdout
```

```
ubuntu@ubuntu:~$ cat file1
```

```
stdout
```

```
ubuntu@ubuntu:~$ cat file2
```

```
stderr
```

- перенаправив стандартный вывод и стандартный канал ошибок в файл 3;

```
ubuntu@ubuntu:~$ sh myscript > file3 && sh myscript 2>> file3
```

```
stderr
```

```
stdout
```

```
ubuntu@ubuntu:~$ cat file3
```

```
stdout
```

```
stderr
```

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

```
ubuntu@ubuntu:~$ sh myscript > file4 && sh myscript 2>> file4
```

```
stderr
```

```
stdout
```

```
ubuntu@ubuntu:~$ cat file4
```

```
stdout
```

```
stderr
```

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

```
ubuntu@ubuntu:~$ sort -r /etc/group | tail -10 | head -3 | tail -1
```

```
cdrom:x:24:ubuntu
```

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

```
ubuntu@ubuntu:~$ ls -l /dev | grep '^b' | wc -l
```

```
18
```

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 (test -n $1) do
```

```
echo $1
```

```
shift
```

```
done
```

```
ubuntu@ubuntu:~$ ./myscript Hello how are you
Hello
how
are
you
```

```
ubuntu@ubuntu:~$ sh myscript Hello world
Hello
world
```

7. Написать скрипт согласно индивидуальному заданию. Номер варианта согласовать с преподавателем.

Вариант 6

Реализовать Меню из двух пунктов: 1-ый пункт – определить количество запущенных данным пользователем процессов bash (предусмотреть ввод имени пользователя); 2-ой пункт – завершить все процессы bash данного пользователя.

```
#!/bin/bash
```

```
clear
```

```
echo "a-kolichestvo processov bash"
```

```
echo "b-zavershit vse processy bash"
```

```
read punkt
```

```
case "$punkt" in
```

```
"a" )
```

```
clear
```

```
echo "Vvedite imya polzovatelya"
```

```
read usr
```

```
ps | grep bash
```

```
pgrep -u $usr bash
```

```
::
```

```
"b" )
```

```
clear
```

```
echo "zavershit vse processy bash"
```

```
kill -9 bash
```

```
::
```

```
* )
```

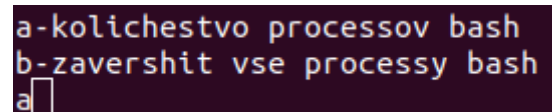
```
clear
```

```
echo "Нет данных."
```

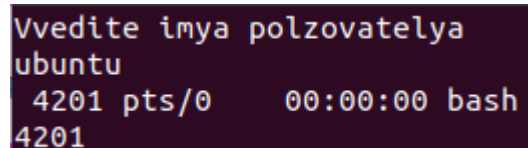
```
::
```

```
esac
```

```
exit 0
```



```
a-kolichestvo processov bash
b-zavershit vse processy bash
a
```



```
Vvedite imya polzovatelya
ubuntu
  4201 pts/0    00:00:00 bash
4201
```