

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

Лабораторная работа №3
за IV семестр
по дисциплине: "Операционные системы и системное программирование"
Тема: "BASH: потоки данных. Программирование"

Выполнил:
студент 2 курса
факультета ЭИС
группы ПО-4 (1)
Галанин П. И. (зачётка №190333)

Проверил:
ст. преподаватель
Давидюк Ю. И.

Лабораторная работа

Тема: "BASH: потоки данных. Программирование".

Цель:

Ход работы:

Задание 1 Условие:

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 >>

Объяснить результаты.

Решение:

Bash

```
echo "Message1" > 1_file-1.txt
cat 1_file-1.txt
```

Out

Message1

Bash

```
echo "Message2" >> 1_file-2.txt
cat 1_file-2.txt
```

Out

Message2

Bash

```
echo "Message3" > 1_file-1.txt
cat 1_file-1.txt
```

Out

Message3

Bash

```
echo "Message4" >> 1_file-2.txt
cat 1_file-2.txt
```

					ЛР.ПО4.190333-03 81 00						
Изм	Лист	№ докум.	Подп.	Дата	Лабораторная работа BASH: потоки данных. Программирование			Лит.	Лист	Листов	
Разраб.	Галанин										
Пров.	Давидюк							Л		2	13
								БрГТУ			
Н. контр.											
Утв.											

Message2

Message4

Вывод:

- Если нет файла, то символ больше (>) создаст файл с сообщением.
- Если нет файла, то два символа больше (>>) создадут файл с сообщением.
- Если файл существует, то символ больше (>) перезапишет файл.
- Если файл существует, то два символа больше (>>) добавить новую строку в файл.

Задание 2 Условие:

Переадресовать стандартный ввод для команды cat на файл.

Решение:

Bash

```
echo "Text..." > 2_file.txt
cat < 2_file.txt
```

Out

Text...

Задание 3 Условие:

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)
- перенаправив стандартный вывод в файл, просмотреть содержимое файла (sh myscript > file1)
- перенаправить стандартный канал ошибок в существующий и несуществующий файлы с помощью символов > и >>
- перенаправив стандартный вывод в файл 1, стандартный канал ошибок - в файл 2
- перенаправив стандартный вывод и стандартный канал ошибок в файл 3
- перенаправив стандартный вывод в файл 4 с помощью символа >, а стандартный канал ошибок в файл 4 с помощью символа >>

Объяснить результаты.

Решение:

Создаем файл.

Bash

```
nano myscript
```

Пишем текст.

Bash

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

Сохраняем: **Ctrl + X, Y, Enter.**

Запускаем скрипт.

Bash

```
sh myscript
```

Out

```
stdout  
stderr
```

Перенаправляем стандартный вывод в файл.

Bash

```
sh myscript > file1
```

Out

```
stderr
```

Просматриваем содержимое файла.

Bash

```
cat file1
```

Out

```
stdout
```

Перенаправляем стандартный поток в несуществующий файл с помощью символа больше >.

Bash

```
sh myscript > 3_file-1.txt
```

Out

```
stderr
```

_____ Bash _____
cat 3_file-1.txt

_____ Out _____
stdout

Перенаправляем стандартный поток в несуществующий файл с помощью двух символов больше >>.

_____ Bash _____
sh myscript >> 3_file-2.txt

_____ Out _____
stderr

_____ Bash _____
cat 3_file-2.txt

_____ Out _____
stdout

Перенаправляем стандартный поток в существующий файл с помощью символа больше >.

_____ Bash _____
sh myscript > 3_file-1.txt

_____ Out _____
stderr

_____ Bash _____
cat 3_file-1.txt

_____ Out _____
stdout

Перенаправляем стандартный поток в существующий файл с помощью двух символов больше >>.

_____ Bash _____
sh myscript >> 3_file-2.txt

_____ Out _____
stderr

_____ Bash _____
cat 3_file-2.txt

_____ Out _____
stdout
stdout

Перенаправляем стандартный вывод в файл 1.

Bash

```
sh myscript 1>1.txt
```

Out

```
stderr
```

Bash

```
cat 1.txt
```

Out

```
stdout
```

Перенаправляем стандартный канал ошибок в файл 2.

Bash

```
sh myscript 2>2.txt
```

Out

```
stdout
```

Bash

```
cat 2.txt
```

Out

```
stderr
```

Перенаправляем стандартный вывод и стандартный канал ошибок в файл 3.

Bash

```
sh myscript 1>3.txt 2>>3.txt  
cat 3.txt
```

Out

```
stdout  
stderr
```

Перенаправляем стандартный вывод в файл 4 с помощью символа >, а стандартный канал ошибок в файл 4 с помощью символа >>.

Bash

```
sh myscript 1>4.txt 2>>4.txt  
cat 4.txt
```

Out

```
stdout  
stderr
```

Изм	Лист	№ докум.	Подп.	Дата

ЛР.ПО4.190333-03 81 00

Задание 4 Условие:

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

Решение:

Выводим отсортированные последние 10 строк.

Bash

```
sort -r /etc/group | tail -n 10
```

Out

```
crontab:x:105:
colord:x:128:
cdrom:x:24:pavel-innokentevich-galanin
bluetooth:x:112:
bin:x:2:
backup:x:34:
avahi:x:124:
avahi-autoipd:x:117:
audio:x:29:pulse
adm:x:4:syslog,pavel-innokentevich-galanin
```

От отсортированных последних 10-ть строк выводим 3-и первые строки.

Bash

```
sort -r /etc/group | tail -n 10 | head -n 3
```

Out

```
crontab:x:105:
colord:x:128:
cdrom:x:24:pavel-innokentevich-galanin
```

От 3-ех строк вывожу последнюю.

Bash

```
sort -r /etc/group | tail -n 10 | head -n 3 | tail -n 1
```

Out

```
cdrom:x:24:pavel-innokentevich-galanin
```

Задание 5 Условие:

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

Решение:

Вывожу количество блочных устройств.

Bash

```
ls -l /dev | grep ^b | wc -l
```

33

Out

Вывожу количество символьных устройств.

```
ls -l /dev | grep ^c | wc -l
```

Out

180

Задание 6 Условие:

Переадресовать стандартный ввод для команды cat на файл.

Решение:

Создаем файл.

```
nano myscript
```

Bash

Пишем sh-скрипт.

```
#!/bin/bash
for i in "$@"
do
    echo "$i"
done
exit 0
```

Bash

Сохраняем нажатиями клавиш **Ctrl + X, Y, Enter**.

Запускаем скрипт через sh.

```
sh myscript a b c d e f
```

Bash

Out

```
a
b
c
d
e
f
```

Запускаем скрипт как файл.

```
./myscript a b c d e f
```

Bash

Изм	Лист	№ докум.	Подп.	Дата

ЛР.ПО4.190333-03 81 00

Лист

8

Out

```
bash: ./myscript: Permission denied
```

Нет прав на исполнение. Добавим права - и запустим скрипт.

Bash

```
ls -l myscript
```

Out

```
-rw-rw-r-- 1 pavel-innokentevich-galanin pavel-innokentevich-galanin 83 Mar  2 03:59 myscript
```

Bash

```
chmod a+x myscript
```

```
ls -l myscript
```

Out

```
-rwxrwxr-x 1 pavel-innokentevich-galanin pavel-innokentevich-galanin 83 Mar  2 03:59 myscript
```

Запускаем скрипт.

Bash

```
./myscript a b c d e f
```

Out

```
a
b
c
d
e
f
```

Задание 7 Условие:

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

Вариант 5

Реализовать меню из трех пунктов (в цикле):

- 1) поиск файла в каталоге. <Имя файла> и <Имя каталога> вводятся пользователем с клавиатуры;
- 2) копирование одного файла в другой каталог. <Имя файла> и <Имя каталога> вводятся пользователем с клавиатуры;
- 3) завершение.

Решение:

					ЛР.ПО4.190333-03 81 00	Лист
Изм	Лист	№ докум.	Подп.	Дата		9

```

1  #!/bin/bash
2
3  function clear_line {
4      echo ""
5  }
6
7  function menu {
8      clear_line
9      echo "Menu:"
10     echo "1. Find file in directory"
11     echo "2. Copy file to directory"
12     echo "3. Exit"
13     clear_line
14 }
15
16 function find_file_in_folder {
17     echo "alpha"
18
19     echo "find <path> -name <whatFind>"
20     clear_line
21
22     echo "path (the path to search in) ="
23     read path
24     clear_line
25
26     echo "whatFind ="
27     read whatFind
28     clear_line
29
30     echo "find $path -name $whatFind"
31     find $path -name $whatFind
32     clear_line
33 }
34
35 function copy_file_to_directory {
36     echo "cp <path1> <path2>"
37     clear_line
38
39     echo "path1 (the path to the file, which you copy) ="

```

```

40     read path1
41     clear_line
42
43     echo "path2 (the path to the folder, to which you want to copy) ="
44     read path2
45     clear_line
46
47     echo "cp $path1 $path2"
48     cp $path1 $path2
49     clear_line
50
51     echo "ls $path2"
52     ls $path2
53     clear_line
54 }
55
56 function main {
57     clear
58     while true
59     do
60         menu
61         read choosenKey
62         clear
63         case $choosenKey in
64             '1')
65                 find_file_in_folder
66                 ;;
67             '2')
68                 copy_file_to_directory
69                 ;;
70             '3')
71                 break
72                 ;;
73             esac # end case
74     done # end while
75 }
76
77 main
78 exit 0

```

Запускаю скрипт

Out

```
./task-7.sh
```

Out

Menu:

1. Find file in directory
2. Copy file to directory
3. Exit

1

Выбираю первый пункт меню

Out

alpha

```
find <path> -name <whatFind>
```

path (the path to search in) =

/home/pavel-innokentevich-galanin/Downloads

whatFind =

*.pdf

```
find /home/pavel-innokentevich-galanin/Downloads -name *.pdf
```

/home/pavel-innokentevich-galanin/Downloads/Telegram Desktop/ПО-4.pdf

/home/pavel-innokentevich-galanin/Downloads/Telegram Desktop/EVMiS_Latex.pdf

Menu:

1. Find file in directory
2. Copy file to directory
3. Exit

2

Ввожу путь /home/pavel-innokentevich-galanin/Downloads.

Ввожу что искать *.pdf.

Получаю результат - и появляется меню.

Выбираю второй пункт меню.

Изм	Лист	№ докум.	Подп.	Дата

ЛР.ПО4.190333-03 81 00

Лист

12

```
cp <path1> <path2>
```

path1 (the path to the file, which you copy) =
/etc/group

path2 (the path to the folder, to which you want to copy) =
/home/pavel-innokentevich-galanin/Disk-O

```
cp /etc/group /home/pavel-innokentevich-galanin/Disk-O
```

```
ls /home/pavel-innokentevich-galanin/Disk-O
```

group

Menu:

1. Find file in directory
2. Copy file to directory
3. Exit

3

Ввожу путь откуда копировать /etc/group.

Ввожу путь куда копировать /home/pavel-innokentevich-galanin/Disk-O.

Получаю результат копирования - и выводится меню.

В меню выбираю пункт 3. Программа завершена.