

Отчет по лабораторной работе №11

Безрук Мария Андреевна

Изучить основы программирования в оболочке ОС UNIX/Linux.
Научиться пи-сать небольшие командные файлы.

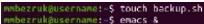
1. Ознакомиться с теоретическим материалом.
2. Изучить основы программирования в оболочке ОС UNIX/Linux.
3. Выполнить упражнения.
4. Ответить на контрольные вопросы.

- 1) Для начала я изучила команды архивации, используя команды «man zip», «man bzip2», «man tar»

```
mmbezruk@username:~$ man zip  
mmbezruk@username:~$ man bzip2  
mmbezruk@username:~$ man tar
```

Figure 1: Команды архивации

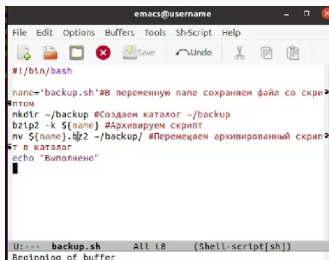
- 2) Далее я создала файл, в котором буду писать первый скрипт, и открыла его в редакторе emacs, используя клавиши «Ctrl-x» и «Ctrl-f» (команды «touch backup.sh» и «emacs &»)



```
mbezruk@username:~$ touch backup.sh  
mbezruk@username:~$ emacs &
```

Figure 2: Создание файла

- 3) После написала скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в вашем домашнем каталоге. При этом файл должен архивироваться одним из архиваторов на выбор zip, bzip2 или tar. При написании скрипта использовала архиватор bzip2.



```
emacs@username
File Edit Options Buffers Tools Sh-Script Help
[Icons: New, Open, Save, Undo, Redo, Find, Print]
#!/bin/bash
name='backup.sh'#В переменную name сохраняем файл со скриптом
mkdir ~/backup #Создаем каталог ~/backup
bzip2 -k ${name} #Архивируем скрипт
mv ${name}.bz2 ~/backup/ #Перемещаем архивированный скрипт в каталог
echo "Выполнено"
```

U:~: backup.sh All 18 (Shell-script[sh])
Beginning of buffer

Figure 3: Первый скрипт

- 4) Проверила работу скрипта (команда «./backup.sh»), предварительно добавив для него право на выполнение (команда «chmod+x*.sh»). Проверила, появился ли каталог backup/, перейдя в него (команда «cdbackup/»), посмотрела его содержимое (команда «ls») и просмотрела содержимое архива (команда «bunzip2 -cbackup.sh.bz2»). Скрипт работает корректно.
- 5) Создала файл, в котором буду писать второй скрипт, и открыла его в редакторе emacs, используя клавиши «Ctrl-x» и «Ctrl-f» (команды «touchprog2.sh» и «emacs&»)

- 6) Написала пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов.

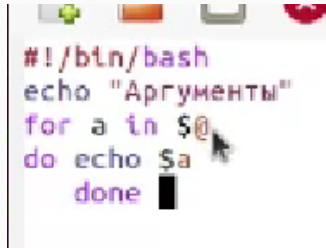
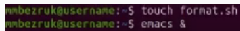
A screenshot of a terminal window with a light gray background. The terminal shows a shell script with the following lines: `#!/bin/bash`, `echo "Аргументы"`, `for a in $@`, `do echo $a`, and `done`. The text is color-coded: the shebang and `done` are purple, `echo` is green, and `for`, `a`, `in`, and `$a` are blue. A mouse cursor is positioned over the `$@` in the `for` line. The terminal window has standard OS window controls (minimize, maximize, close) at the top.

Figure 4: Второй скрипт

- 7) Проверила работу написанного скрипта (команды «./prog2.sh 0 1 2 3 4» и «./prog2.sh 0 1 2 3 4 5 6 7 8 9 10 11»), предварительно добавив для него право на выполнение (команда «chmod+x*.sh»). Вводила аргументы, количество которых меньше 10 и больше 10. Скрипт работает корректно.
- 8) Создала файл, в котором буду писать третий скрипт, и открыла его в редакторе emacs, используя клавиши «Ctrl-x» и «Ctrl-f» (команды «touchprogl.sh» и «emacs&»)

- 9) Написала командный файл -аналог команды ls (без использования самой этой команды и команды dir). Он должен выдавать информацию о нужном каталоге и выводить информацию о возможностях доступа к файлам этого каталога
- 10) Далее проверила работу скрипта (команда «./proglsh.sh»), предварительно добавив для него право на выполнение (команда «chmod+x*.sh»).Скрипт работает корректно.

- 11) Для четвертого скрипта также создала файл (команда «touchformat.sh»)и открыла его в редакторе emacs, используя клавиши «Ctrl-x» и «Ctrl-f» (команда «emacs&»)



```
mmbezruk@username:~$ touch format.sh  
mmbezruk@username:~$ emacs &
```

Figure 5: Создание файла

- 12) 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

13) Проверила работу написанного скрипта (команда «./format.sh~pdfshtxt.doc»), предварительно добавив для него право на выполнение (команда «chmod+x*.sh»), а также создав дополнительные файлы с разными расширениями (команда «touchfile.pdf file1.doc file2.doc»).Скрипт работает корректно.

[illegible]

Figure 6: Проверка работы скрипта

В ходе выполнения данной лабораторной работы я изучила основы программирования в оболочке ОС UNIX/Linux и научилась писать небольшие командные файлы.