

# Презентация по лабораторной работе

---

Elizaveta Savchenko

02.06.2021

<sup>1</sup>RUDN University, Moscow, Russian Federation

## Цель работы

---

Изучить основы программирования в оболочке ОС UNIX, научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

## Выполнение лабораторной работы

---

1. Написала командный файл, реализующий упрощённый механизм семафоров. Командный файл должен в течение некоторого времени  $t_1$  дожидаться освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использовать его в течение некоторого времени  $t_2 < t_1$ , также выдавая информацию о том, что ресурс используется соответствующим командным файлом (процессом). Запустила командный файл в одном виртуальном терминале в фоновом режиме, перенаправив его вывод в другой (`> /dev/tty#`, где `#` — номер терминала куда перенаправляется вывод), в котором также запущен этот файл, но не фоновом, а в привилегированном режиме. Доработать программу так, чтобы имелась возможность взаимодействия трёх и более процессов.

```
#!/bin/bash
lockfile="./lockfile"
exec {fn}>$lockfile
echo "lock"
until flock -n ${fn}
do
    echo "not lock"
    sleep 1
    flock -n ${fn}
done
for ((i=0;i<=5;i++))
do
    echo "work"
    sleep 1
done
```

--- lab10.sh All L1 (Shell-script[bash]) Ср июн 2 11:09 0.32

```
ensavchenko@dk8n62 ~ $ emacs
ensavchenko@dk8n62 ~ $ touch lab10.sh
ensavchenko@dk8n62 ~ $ chmod +x lab10.sh
ensavchenko@dk8n62 ~ $ ./lab10.sh
lock
work
work
work
work
work
work
ensavchenko@dk8n62 ~ $
```

2. Реализовала команду `man` с помощью командного файла. Изучила содержимое каталога `/usr/share/man/man1`. В нем находятся архивы текстовых файлов, содержащих справку по большинству установленных в системе программ и команд. Каждый архив можно открыть командой `less` сразу же просмотрев содержимое справки. Командный файл должен получать в виде аргумента командной строки название команды и в виде результата выдавать справку об этой команде или сообщение об отсутствии справки, если соответствующего файла нет в каталоге `man1`.



```
#!/bin/bash  
cd /usr/share/man/man1  
less $1*
```

-- lab101.sh All L1 (Shell-script[bash]) Ср июн 2 11:13 0.35

```
ensavchenko@dk8n62 ~ $ touch lab101.sh
ensavchenko@dk8n62 ~ $ chmod +x lab101.sh
ensavchenko@dk8n62 ~ $ ./lab101.sh less
ensavchenko@dk8n62 ~ $
```

```

LESS(1)                                General Commands Manual                LESS(1)

NAME

    less - opposite of more

SYNOPSIS

    less -?
    less --help
    less -V
    less --version
    less [-[+]aAbcCdEfFgGiIJKLmMnNqQrRsSuUVwWx~]
        [-b space] [-h lines] [-j line] [-k Keyfile]
        [-{oO} logfile] [-p pattern] [-P prompt] [-t tag]
        [-T tagsfile] [-x tab,...] [-y lines] [-{z} lines]
        [-# shift] [+{+}cmd] [--] [filename]...
    (See the OPTIONS section for alternate option syntax with long option names.)

DESCRIPTION

    Less is a program similar to more(1), but which allows backward movement in the file as well as forward movement. Also, less does not have to read the entire input file before starting, so with large input files it starts up faster than text editors like vi(1). Less uses termcap (or terminfo on some systems), so it can run on a variety of terminals. There is even limited support for hardcopy terminals. (On a hardcopy terminal, lines which should be printed at the top of the screen are prefixed with a caret.)

    Commands are based on both more and vi. Commands may be preceded by a decimal number, called N in the descriptions below. The number is used by some commands, as indicated.

COMMANDS

    In the following descriptions, ^X means control-X. ESC stands for the ESCAPE key; for example ESC-v means the two character sequence "ESCAPE", then "v".

    h or H Help: display a summary of these commands. If you forget all the other commands, remember this one.

    SPACE or ^V or f or ^F
        Scroll forward N lines, default one window (see option -z below). If N is more than the screen size, only the final screenful is displayed. Warning: some systems use ^V as a special literalization character.

    z
        Like SPACE, but if N is specified, it becomes the new window size.

    ESC-SPACE
        Like SPACE, but scrolls a full screenful, even if it reaches end-of-file in the process.

    ENTER or RETURN or ^N or e or ^E or j or ^J
        Scroll forward N lines, default 1. The entire N lines are displayed, even if N is more than the screen size.

```

```

Файл  Правка  Вид  Закладки  Настройка  Справка

411toppm User Manual(0)                                411toppm User Manual(0)

NAME
    411toppm - convert Sony Mavica .411 image to PPM

SYNOPSIS
    411toppm [-width width] [-height height] [411file]

DESCRIPTION
    This program is part of Netpbm(1)

    411toppm reads a .411 file, such as from a Sony Mavic camera, and converts it to a PPM image as
    output.

    Output is to Standard Output.

    The originator of this program and decipherer of the .411 format, Steve Allen <sla@alumni.cal-
    tech.edu>, has this to say about the utility of this program: 'There's so little image in a 64x48
    thumbnail (especially when you have the full size JPG file) that the only point in doing this was
    to answer the implicit challenge posed by the manual stating that only the camera can use these
    files.'

OPTIONS
    All options may be abbreviated to the shortest unique prefix.

    -width The width (number of columns) of the input image. Default is 64.

    -height
        The height (number of rows) of the input image. Default is 48.

SEE ALSO
    ppm(5)

DOCUMENT SOURCE
    This manual page was generated by the Netpbm tool 'makeman' from HTML source. The master docu-
    mentation is at

        http://netpbm.sourceforge.net/doc/411toppm.html

netpbm documentation                                03 March 2001                                411toppm User Manual(0)
411toppm.1.bz2 (file 1 of 5106) lines 1-49/49 (END) - Next: 7-1.bz2

```

3. Используя встроенную переменную `$RANDOM`, написала командный файл, генерирующий случайную последовательность букв латинского алфавита. Учла, что `$RANDOM` выдаёт псевдослучайные числа в диапазоне от 0 до 32767.

```
#!/bin/bash
M=10
c=1
d=1
echo
echo "10 random words:"
while (($c!=($M+1)))
do
    echo $(for((i=1;i<=10;i++)); do printf '%s' "${RANDOM:0:1}"; done) | tr '0-9' ' [a-z] '
    echo $d
    ((c+=1))
    ((d+=1))
done
```

```
ensavchenko@dk8n62 ~ $ touch lab100.sh
ensavchenko@dk8n62 ~ $ chmod +x lab100.sh
ensavchenko@dk8n62 ~ $ ./lab100.sh
```

10 random words:

bbaacgbcba

1

aabahcaaba

2

babaabacbc

3

baabfdbfbc

4

ebbfaaabga

5

aaacabbabc

6

iaibbbadeh

7

bfbdbcbbba

8

bdfaacibhd

9

aaecbhcdba

10

## Выводы

---



Я изучила основы программирования в оболочке ОС UNIX, научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.