

Операционные системы

Программирование в командном процессоре ОС UNIX.

Виктория Шангина

22 апреля 2025

Российский университет дружбы народов, Москва, Россия

Цели и задачи работы

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

1 Выполнить 3 задания

Процесс выполнения лабораторной работы

1. Написали командный файл, реализующий упрощённый механизм семафоров.
Командный файл в течение некоторого времени t_1 дожидается освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использует его в течение некоторого времени $t_2 < t_1$, также выдавая информацию о том, что ресурс используется соответствующим командным файлом (процессом).

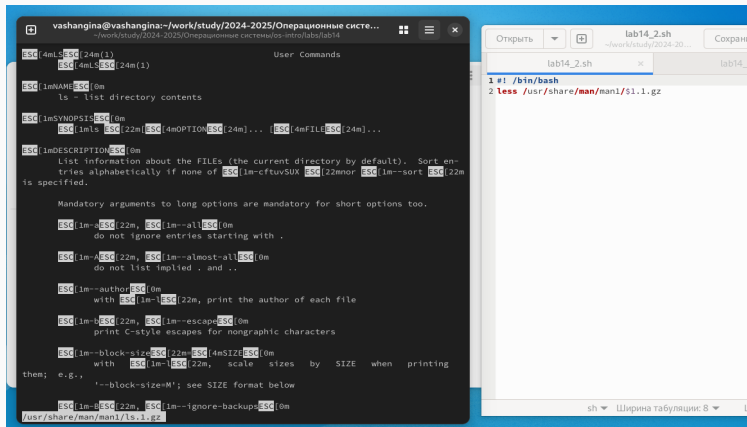
Выполнение работы

The screenshot displays two terminal windows side-by-side. The left window shows a user named vashangina at a machine named vashangina, working in the directory ~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14. The user has executed the command `chmod +x lab14_*`. They then run `./lab14_1.sh &`, which outputs [1] 3495. Subsequently, they execute `./lab14_1.sh`, which repeatedly prints "Пишу в файл..." (Writing to file...). After running `./lab14_1.sh` again, it prints "Жду разблокировки файла" (Waiting for file unlock). The right window shows the contents of the `lab14_1.sh` script. It starts with `#!/bin/bash` and contains a loop that checks if a lockfile exists. If it does, it sleeps for 1 second and echoes "Жду разблокировки файла". If no lockfile exists, it creates one (`touch lockfile`), sets `c=10`, enters another loop where it echoes "Записываем в файл..." (Writing to file...) until `c=0`, and finally removes the lockfile (`rm lockfile`). The script ends with a `done` statement.

Рис. 1: Задание 1

2. Реализовали команду `man` с помощью командного файла. Изучили содержимое каталога `/usr/share/man/man1`. В нем находятся архивы текстовых файлов, содержащих справку по большинству установленных в системе программ и команд.

Выполнение работы



The image shows a terminal window on the left and a file editor on the right. The terminal window, titled 'vashangina@vashangina:~/work/study/2024-2025/Операционные систе...', displays the manual page for the 'ls' command. The file editor, titled 'lab14_2.sh', shows the first two lines of a script: '1 #! /bin/bash' and '2 less /usr/share/man/man1/\$1.1.gz'.

```
vashangina@vashangina:~/work/study/2024-2025/Операционные систе...
~/work/study/2024-2025/Операционные системы/ос-intro/labs/lab14

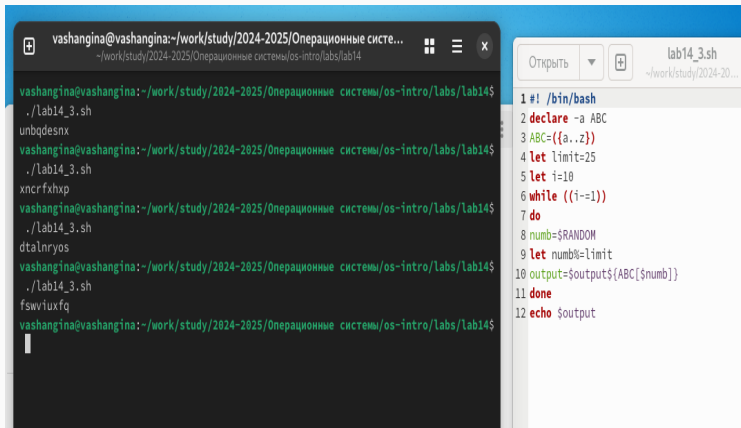
ESC[4mLESC[24m(1) User Commands
ESC[4mLESC[24m(1)
ESC[1mNAMEESC[0m
ls - list directory contents
ESC[1mSYNOPSISESC[0m
ESC[1mLs ESC[22m[ESC[4mOPTIONESC[24m]... [ESC[4mFILEESC[24m]...
ESC[1mDESCRIPTIONESC[0m
List information about the FILES (the current directory by default). Sort en-
tries alphabetically if none of ESC[1m-cftuvSUX ESC[22mnor ESC[1m--sort ESC[22m
is specified.
Mandatory arguments to long options are mandatory for short options too.
ESC[1m-aESC[22m, ESC[1m--allESC[0m
do not ignore entries starting with .
ESC[1m-AESC[22m, ESC[1m--almost-allESC[0m
do not list implied . and ..
ESC[1m--authorESC[0m
with ESC[1m-lESC[22m, print the author of each file
ESC[1m-bESC[22m, ESC[1m--escapeESC[0m
print C-style escapes for nongraphic characters
ESC[1m--block-sizeESC[22m[ESC[4mSIZEESC[0m
with ESC[1m-lESC[22m, scale sizes by SIZE when printing
them; e.g.,
'--block-size=M'; see SIZE format below
ESC[1m-BESC[22m, ESC[1m--ignore-backupsESC[0m
/usr/share/man/man1/ls.1.gz

lab14_2.sh
~/work/study/2024-20...
lab14_2.sh
1 #! /bin/bash
2 less /usr/share/man/man1/$1.1.gz
sh Ширина табуляции: 8
```

Рис. 2: Задание 2

3. Используя встроенную переменную `$RANDOM`, написали командный файл, генерирующий случайную последовательность букв латинского алфавита

Выполнение работы



The image shows a terminal window on the left and a file editor on the right. The terminal window has a title bar with the text "vashangina@vashangina:~/work/study/2024-2025/Операционные систе..." and a subtitle "vashangina@vashangina:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14". The terminal content shows the user running the script "lab14_3.sh" multiple times, with various random strings appearing as output. The file editor on the right has a title bar with "lab14_3.sh" and a subtitle "vashangina@vashangina:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14". The file content is a shell script with 12 lines.

```
vashangina@vashangina:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
unbqdesnx
vashangina@vashangina:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
xncrfxhxp
vashangina@vashangina:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
dtalnryos
vashangina@vashangina:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
fswviuxfq
vashangina@vashangina:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$
```

```
1 #!/bin/bash
2 declare -a ABC
3 ABC=({a..z})
4 let limit=25
5 let i=10
6 while ((i-=1))
7 do
8     num=${RANDOM}
9     let num%=limit
10    output=${ABC[$num]}
11 done
12 echo $output
```

Рис. 3: Задание 3

Выводы по проделанной работе

Изучили основы программирования в оболочке ОС UNIX. Научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.