

Операционные системы

Программирование в командном процессоре ОС UNIX.

Скрипник Виктория Дмитриевна

31 мая 2025

Российский университет дружбы народов, Москва, Россия

Цели и задачи работы

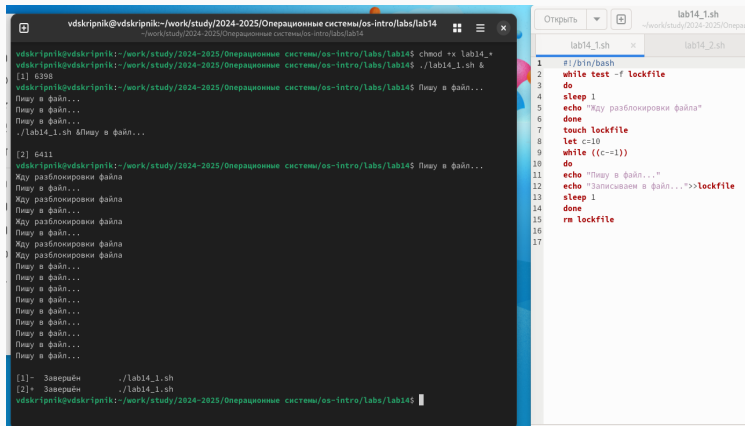
Изучить основы программирования в оболочке ОС UNIX. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов

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 image shows a terminal window on the left and a code editor on the right. The terminal window displays the execution of a script named `lab14_1.sh`. The script sets permissions for `lab14_*` files and then runs `./lab14_1.sh &`. The output shows the script running in the background, printing "Пишу в файл..." (Writing to file...) multiple times. The code editor on the right shows the content of `lab14_1.sh`, which is a shell script that creates a lockfile, writes to it, and then removes it.

```
vdskrpnik@vdskrpnik:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ chmod +x lab14_*
vdskrpnik@vdskrpnik:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_1.sh &
[1] 6398
vdskrpnik@vdskrpnik:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ Пишу в файл...
Пишу в файл...
Пишу в файл...
./lab14_1.sh & Пишу в файл...

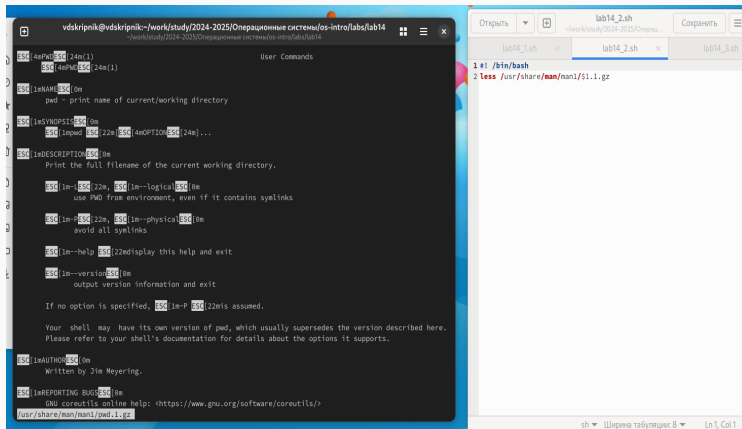
[2] 6411
vdskrpnik@vdskrpnik:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
Пишу в файл...
[1]- Завершён      ./lab14_1.sh
[2]+ Завершён      ./lab14_1.sh
vdskrpnik@vdskrpnik:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$
```

```
lab14_1.sh
1 #!/bin/bash
2 while test -f lockfile
3 do
4     sleep 1
5     echo "Жду разблокировки файла"
6 done
7 touch lockfile
8 let c-=10
9 while ((c-=1))
10 do
11     echo "Пишу в файл..."
12     echo "Записываем в файл...">>lockfile
13     sleep 1
14 done
15 rm lockfile
16
17
```

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



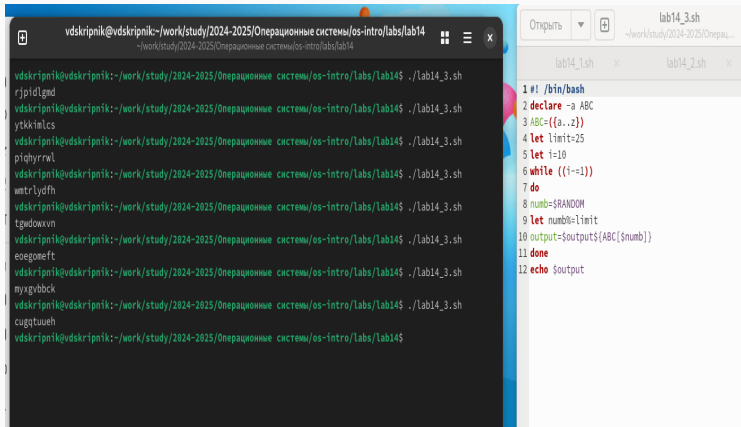
The image shows two windows from a Raspberry Pi desktop environment. The left window is a terminal titled 'vdskrprik@vdskrprik:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14'. It displays the help text for the 'pwd' command, which explains its options: --logical (use PWD from environment), --physical (avoid all symlinks), --help (display this help), and --version (output version information). The prompt is at the bottom: `/usr/share/man/man1/pwd.1.gz`. The right window is a file editor titled 'lab14_2.sh' with tabs for 'lab14_1.sh', 'lab14_2.sh' (active), and 'lab14_3.sh'. It contains two lines of code: `1 #! /bin/bash` and `2 less /usr/share/man/man1/$1.1.gz`. The status bar at the bottom right indicates 'sh' and 'Ln 1, Col 1'.

```
vdskrprik@vdskrprik:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14
ESC[4mPwdESC[24m(1)
ESC[4mPwdESC[24m(1)
ESC[1mNAMEESC[0m
pwd - print name of current/working directory
ESC[1mSYNOPSISESC[0m
ESC[1mpwd [22mESC[4mOPTIONESC[24m]...
ESC[1mDESCRIPTIONESC[0m
Print the full filename of the current working directory.
ESC[1m--logicalESC[0m
use PWD from environment, even if it contains symlinks
ESC[1m--physicalESC[0m
avoid all symlinks
ESC[1m--helpESC[0m[22mdisplay this help and exit
ESC[1m--versionESC[0m
output version information and exit
If no option is specified, ESC[1m-PESC[22mis assumed.
Your shell may have its own version of pwd, which usually supersedes the version described here.
Please refer to your shell's documentation for details about the options it supports.
ESC[1mAUTHORESC[0m
Written by Jim Meyering.
ESC[1mREPORTING BUGSESC[0m
GNU coreutils online help: <https://www.gnu.org/software/coreutils/>
/usr/share/man/man1/pwd.1.gz

lab14_2.sh
lab14_1.sh x lab14_2.sh x lab14_3.sh
1 #! /bin/bash
2 less /usr/share/man/man1/$1.1.gz
sh Ширина таблицы: 8 Ln 1, Col 1
```

Рис. 2: Задание 2

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



The image shows a terminal window on the left and a code editor on the right. The terminal window has a title bar with the text "vdskrpnik@vdskrpnik:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14". The terminal content shows a series of commands being executed: `./lab14_3.sh`, followed by random strings like `rjpidlgmd`, `ytkkimlcs`, `piqhyrrwl`, `wntrlydfh`, `tgwdowxvn`, `eoegomeft`, `myxgvbbck`, and `cuggtuueh`. The code editor on the right has a title bar with "lab14_3.sh" and shows the following script content:

```
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   numb=$RANDOM
9   let numb%=limit
10  output=$output${ABC[numb]}
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.