

Операционные системы

Программирование в командном процессоре ОС UNIX.

Андрианов Александр Павлович

26 июня 2025

Российский университет дружбы народов, Москва, Россия

Цели и задачи работы

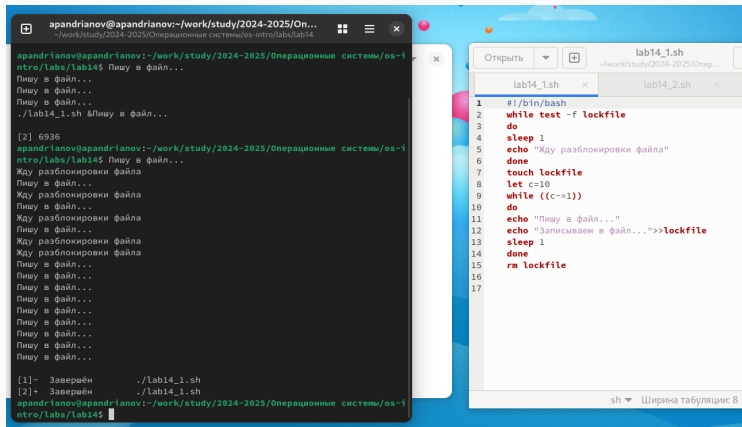
Изучить основы программирования в оболочке ОС UNIX. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов

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 and a script editor. The terminal window on the left displays the execution of a script named `lab14_1.sh`. The script repeatedly attempts to write to a file, but it is blocked because the file is already open for writing by another process. The terminal output shows the script running in the background, and the user is prompted to press Enter to see the output. The script output shows the file being locked and the user being prompted to wait for the file to be unlocked. The script editor on the right shows the source code of `lab14_1.sh`, which is a shell script that creates a lockfile, sleeps for 1 second, and then writes to the file. The script is designed to demonstrate file locking in a multi-user environment.

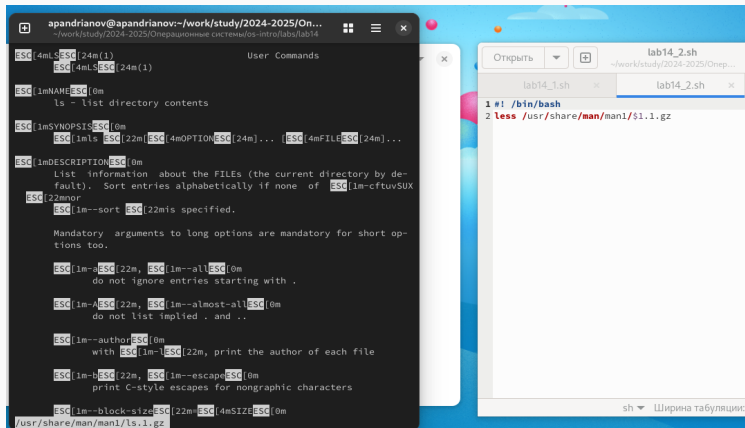
```
apandrianov@apandrianov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_1.sh &Пишу в файл...
[2] 6936
apandrianov@apandrianov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
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
[1]- Завершён      ./lab14_1.sh
[2]+ Завершён      ./lab14_1.sh
apandrianov@apandrianov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$
```

```
lab14_1.sh
#!/bin/bash
while test -f lockfile
do
sleep 1
echo "Жду разблокировки файла"
done
touch lockfile
let c=10
while ((c-=1))
do
echo "Пишу в файл..."
echo "Записываем в файл...">>lockfile
sleep 1
done
rm lockfile
```

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 desktop environment with two windows. The left window is a terminal titled 'apandrianov@apandrianov:~/work/study/2024-2025/On...' displaying the man page for the 'ls' command. The right window is a file manager titled 'lab14_2.sh' showing the contents of a file named 'lab14_1.sh'.

```
apandrianov@apandrianov:~/work/study/2024-2025/On...
~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14

ESC[4mL ESC[24m(1) User Commands
ESC[4mL ESC[24m(1)
ESC[1mNAME ESC[0m
ls - list directory contents
ESC[1mSYNOPSIS ESC[0m
ESC[1mls ESC[22mESC[4mOPTION ESC[24m... ESC[4mFILE ESC[24m...
ESC[1mDESCRIPTION ESC[0m
List information about the FILES (the current directory by default). Sort entries alphabetically if none of ESC[1m-cftuvSUX
ESC[22mnor
ESC[1m--sort ESC[22mis specified.
Mandatory arguments to long options are mandatory for short options too.
ESC[1m-a ESC[22m, ESC[1m--all ESC[0m
do not ignore entries starting with .
ESC[1m-A ESC[22m, ESC[1m--almost-all ESC[0m
do not list implied . and ..
ESC[1m--author ESC[0m
with ESC[1m-l ESC[22m, print the author of each file
ESC[1m-b ESC[22m, ESC[1m--escape ESC[0m
print C-style escapes for nongraphic characters
ESC[1m--block-size ESC[22mESC[4mSIZE ESC[0m
/usr/share/man/man1/ls.1.gz
```

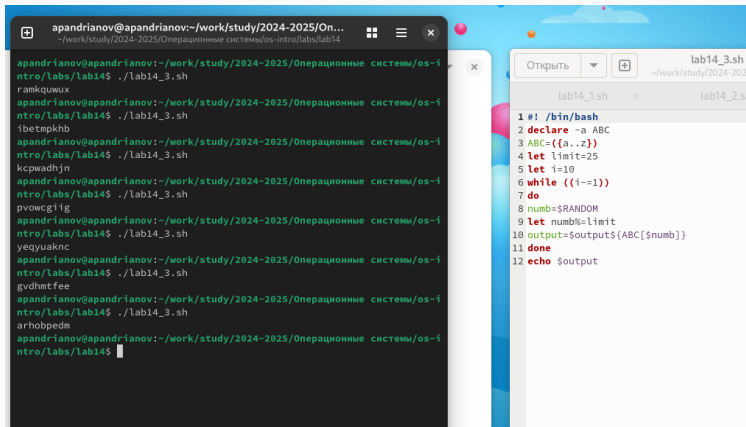
```
lab14_2.sh
~/work/study/2024-2025/Onep...
lab14_1.sh x lab14_2.sh x
1 #! /bin/bash
2 less /usr/share/man/man1/$1.1.gz
```

sh Ширина табуляции:

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 that reads "apandrianov@apandrianov:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14". It displays a series of commands and their outputs. The commands are: `./lab14_3.sh`, `./lab14_3.sh`, `./lab14_3.sh`, `./lab14_3.sh`, `./lab14_3.sh`, `./lab14_3.sh`, `./lab14_3.sh`, `./lab14_3.sh`, `./lab14_3.sh`, `./lab14_3.sh`, and `./lab14_3.sh`. The outputs are: `ramkquwux`, `ibetmpkhh`, `kcpwadhjn`, `pvcwgiig`, `yeqyuaknc`, `gvdhmtfee`, `arhobpeda`, and a blank line. The file editor on the right has a title bar that reads "lab14_3.sh" and shows the following code:

```
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.