

Операционные системы

Программирование в командном процессоре ОС UNIX.

Гаджиев Мирзе Керимович

27 апреля 2025

Российский университет дружбы народов, Москва, Россия

Цели и задачи работы

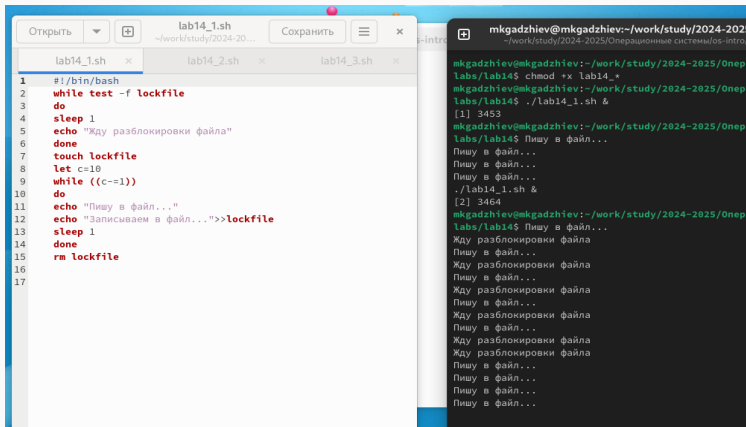
Изучить основы программирования в оболочке ОС UNIX. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов

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



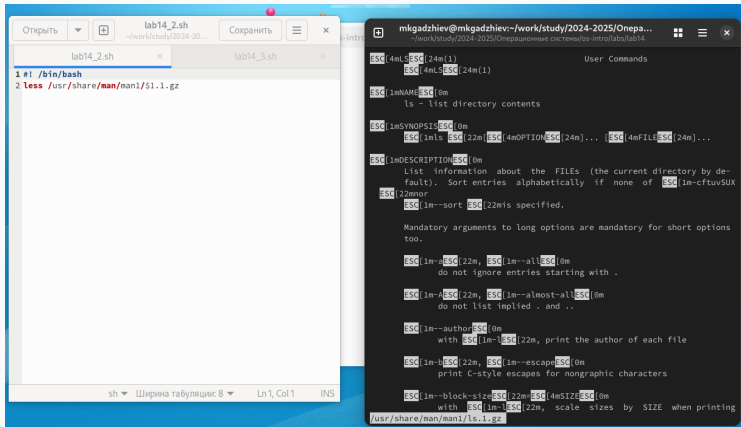
```
lab14_1.sh
1  #!/bin/bash
2  while test -f lockfile
3  do
4      sleep 1
5      echo "Жду разблокировки файла"
6  done
7  touch lockfile
8  let c=10
9  while ((c-->0))
10 do
11     echo "Пишу в файл..."
12     echo "Записываем в файл...">>lockfile
13     sleep 1
14 done
15 rm lockfile
16
17
```

```
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/0nep
labs/lab14$ chmod +x lab14_*
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/0nep
labs/lab14$ ./lab14_1.sh &
[1] 3453
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/0nep
labs/lab14$ Пишу в файл...
Пишу в файл...
Пишу в файл...
./lab14_1.sh &
[2] 3464
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/0nep
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...
```

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



```
lab14_2.sh
~ /work/study/2024-20...
lab14_2.sh x lab14_3.sh x
1 #! /bin/bash
2 less /usr/share/man/man1/ls.1.gz

sh  Ширина табуляции: 8  Ln 1, Col 1  INS
```

```
mkgadzhiev@mkgadzhiev- /work/study/2024-2025/Onepa...
~ /work/study/2024-2025/Операционные системы/os-intro/labs/lab14
User Commands
ESC[4mESC[24m(1)
ESC[4mESC[24m(1)

ESC[1mNAMEESC[0m
ls - list directory contents

ESC[1mSYNOPSISESC[0m
ESC[1mESC[22mESC[4mOPTIONESC[24m]... [ESC[4mFILEESC[24m]...

ESC[1mDESCRIPTIONESC[0m
List information about the FILES (the current directory by default). Sort entries alphabetically if none of ESC[1m-cftuvSUXESC[22mmor
ESC[1m--sortESC[22mis 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

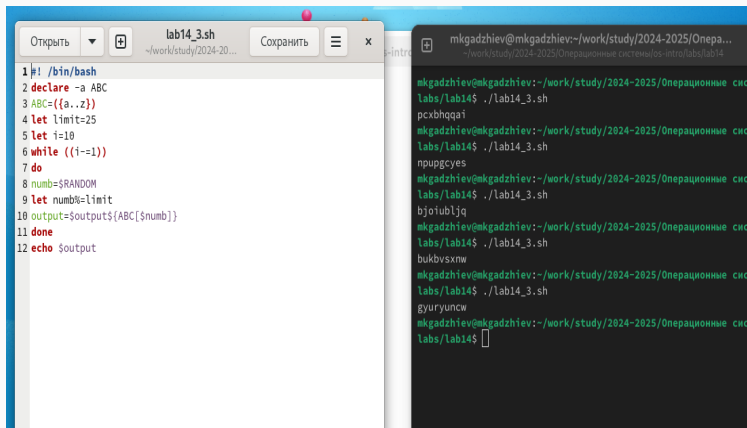
ESC[1m-bESC[22m, ESC[1m--escapeESC[0m
print C-style escapes for nongraphic characters

ESC[1m--block-sizeESC[22mESC[4mSIZEESC[0m
with ESC[1m-bESC[22m, scale sizes by SIZE when printing
/usr/share/man/man1/ls.1.gz
```

Рис. 2: Задание 2

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

Выполнение работы



The image displays two terminal windows side-by-side. The left window, titled 'lab14_3.sh', shows the source code of a shell script. The script declares an array 'ABC' with values 'a..z', sets a 'limit' of 25, and enters a 'while' loop that runs 10 times. In each iteration, it generates a random number, selects an element from the 'ABC' array, and prints it. The right window shows the execution of this script from the 'labs/lab14' directory. It displays the command './lab14_3.sh' being run multiple times, with the output showing a sequence of random characters: 'pcxbhqqai', 'npugcyes', 'bjoiu bljq', 'bukbvsxnw', and 'gyurguncw'.

```
1 #!/bin/bash
2 declare -a ABC
3 ABC=({a..z})
4 let limit=25
5 let i=10
6 while ((i-->0))
7 do
8   numb=$((RANDOM%limit))
9   let numb=numb%limit
10  output=${ABC[numb]}
11 done
12 echo $output
```

```
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
pcxbhqqai
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
npugcyes
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
bjoiu bljq
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
bukbvsxnw
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
gyurguncw
mkgadzhiev@mkgadzhiev:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$
```

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.