

Операционные системы

Программирование в командном процессоре ОС UNIX.

Лабси Мохаммед

19 апреля 2025

Российский университет дружбы народов, Москва, Россия

Цели и задачи работы

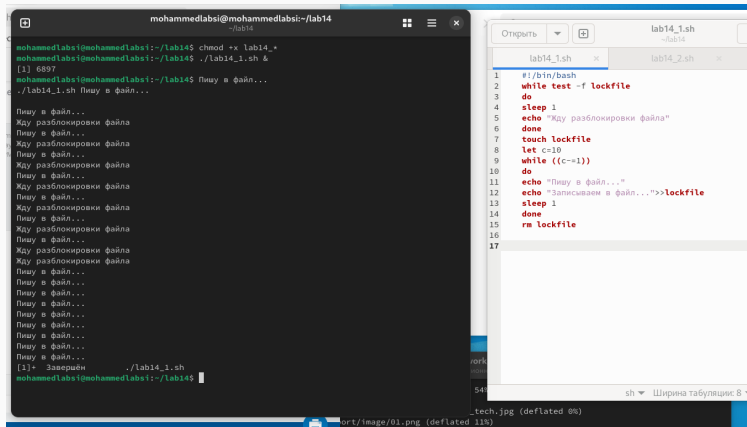
Изучить основы программирования в оболочке ОС UNIX. Научиться писать более сложные командные файлы с использованием логических управляющих конструкций и циклов

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 image shows a terminal window on the left and a script editor on the right. The terminal window displays the execution of a script named `lab14_1.sh`. The script's output consists of a series of messages: "Пишу в файл..." (Writing to file...) followed by "Жду разблокировки файла" (Waiting for file unlock), which is repeated multiple times. The terminal prompt is `mohammedlabsi@mohammedlabsi:~/lab14`. The script editor on the right shows the source code of `lab14_1.sh`. The script is a Bash script that checks if a lockfile exists. If it does, it sleeps for 1 second, echoes "Жду разблокировки файла" (Waiting for file unlock), and then touches the lockfile. If the lockfile does not exist, it echoes "Пишу в файл..." (Writing to file...), echoes "Записываем в файл..." (Writing to file...) followed by `>>>lockfile`, sleeps for 1 second, and then removes the lockfile.

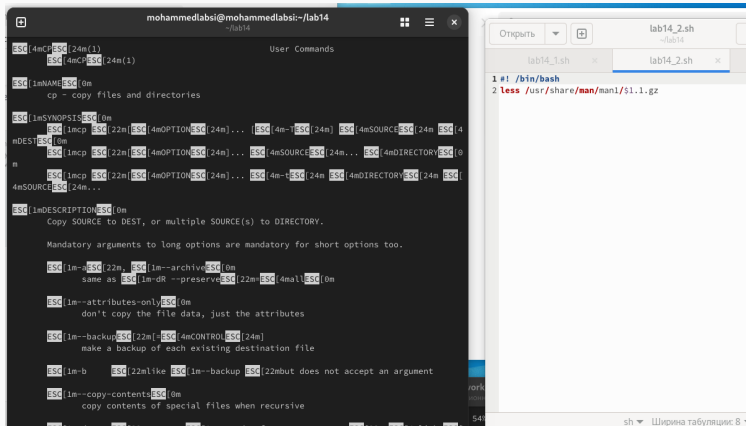
```
mohammedlabsi@mohammedlabsi:~/lab14$ chmod +x lab14_1.sh
mohammedlabsi@mohammedlabsi:~/lab14$ ./lab14_1.sh &
[1] 6897
mohammedlabsi@mohammedlabsi:~/lab14$ Пишу в файл...
./lab14_1.sh Пишу в файл...
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки файла
Пишу в файл...
Жду разблокировки 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...
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
mohammedlabsi@mohammedlabsi:~/lab14$
```

```
#!/bin/bash
1 while test -f lockfile
2 do
3
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

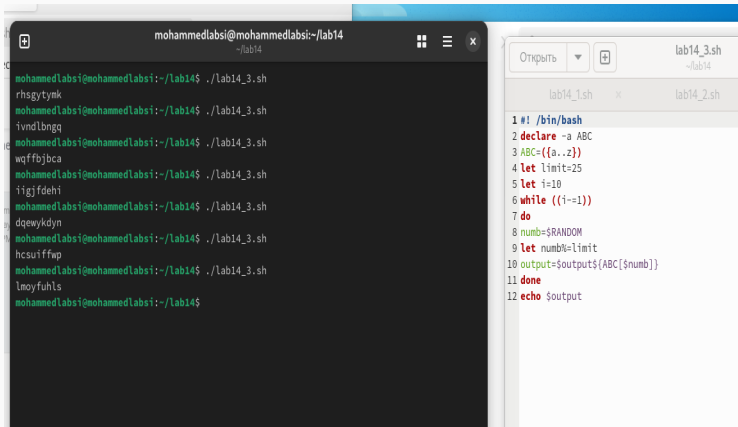


The image shows a terminal window on the left and a file manager window on the right. The terminal window displays the help text for the 'cp' command. The file manager window shows a terminal window with the same help text.

```
mohammedlabsi@mohammedlabsi:~/lab14
ESC[4mCPESC[24m(1)
ESC[4mCPESC[24m(1)
User Commands
ESC[1mNAMEESC[0m
cp - copy files and directories
ESC[1mSYNOPSISESC[0m
ESC[1mcp ESC[22m[ESC[4mOPTIONESC[24m]... ESC[4m-ESC[24m] ESC[4mSOURCEESC[24m ESC[4m
mDESTESC[0m
ESC[1mcp ESC[22m[ESC[4mOPTIONESC[24m]... ESC[4mSOURCEESC[24m... ESC[4mDIRECTORYESC[0m
ESC[1mcp ESC[22m[ESC[4mOPTIONESC[24m]... ESC[4m-ESC[24m ESC[4mDIRECTORYESC[24m ESC[4m
4mSOURCEESC[24m...
ESC[1mDESCRIPTIONESC[0m
Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.
Mandatory arguments to long options are mandatory for short options too.
ESC[1m-aESC[22m, ESC[1m--archiveESC[0m
same as ESC[1m-dR --preserveESC[22mESC[4mallESC[0m
ESC[1m--attributes-onlyESC[0m
don't copy the file data, just the attributes
ESC[1m--backupESC[22m[ESC[4mCONTROLESC[24m]
make a backup of each existing destination file
ESC[1m-b ESC[22mlike ESC[1m--backup ESC[22mbut does not accept an argument
ESC[1m--copy-contentsESC[0m
copy contents of special files when recursive
```

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 script editor on the right. The terminal window, titled 'mohammedlabsi@mohammedlabsi:~/lab14', shows the execution of a script named 'lab14_3.sh' multiple times, resulting in random alphanumeric strings. The script editor, titled 'lab14_3.sh', shows the content of the script, which is a bash script that declares an array 'ABC' containing all lowercase letters, sets a limit of 25, and uses a while loop to generate random strings from the array.

```
mohammedlabsi@mohammedlabsi:~/lab14$ ./lab14_3.sh
rhsgytynk
mohammedlabsi@mohammedlabsi:~/lab14$ ./lab14_3.sh
ivndlbngq
mohammedlabsi@mohammedlabsi:~/lab14$ ./lab14_3.sh
wqffbjbca
mohammedlabsi@mohammedlabsi:~/lab14$ ./lab14_3.sh
iigjfdehi
mohammedlabsi@mohammedlabsi:~/lab14$ ./lab14_3.sh
dqewykdyn
mohammedlabsi@mohammedlabsi:~/lab14$ ./lab14_3.sh
hcsuiffwp
mohammedlabsi@mohammedlabsi:~/lab14$ ./lab14_3.sh
lmoyfuhls
mohammedlabsi@mohammedlabsi:~/lab14$
```

```
1 #! /bin/bash
2 declare -a ABC
3 ABC={a..z}
4 let limit=25
5 let i=10
6 while ((i--))
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.