

Операционные системы

Программирование в командном процессоре ОС UNIX.

Рами Альмансара

3 сентября 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

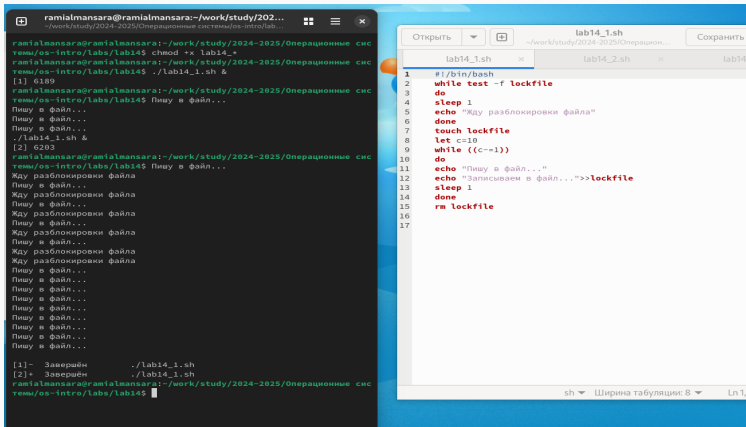



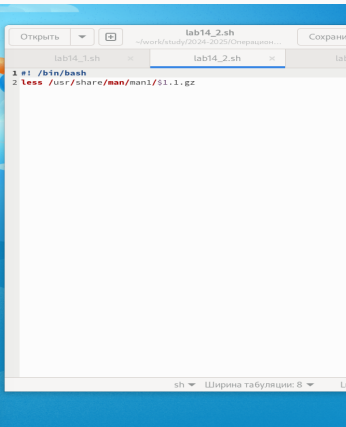
Рис. 1: Задание 1

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



```
ramialmansara@ramialmansara:~/work/study/202...
ESC[4mPwdESC[24m(1) User Commands
ESC[4mPwdESC[24m(1)
ESC[1mNAMEESC[0m
pwd - print name of current/working directory
ESC[1mSYNOPSISESC[0m
ESC[1mpwdESC[22mESC[4mOPTIONESC[24m...
ESC[1mDESCRIPTIONESC[0m
Print the full filename of the current working directory.
ESC[1m-tESC[22m, ESC[1m--logicalESC[0m
use PWD from environment, even if it contains sym-
links
ESC[1m-PESC[22m, ESC[1m--physicalESC[0m
avoid all symlinks
ESC[1m--helpESC[22mESC[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/soft-
ware/coreutils/>
Report any translation bugs to <https://translationpro-
ject.org/team/>
ESC[1mCOPYRIGHTESC[0m
/usr/share/man/man1/pwd.1.gz
```

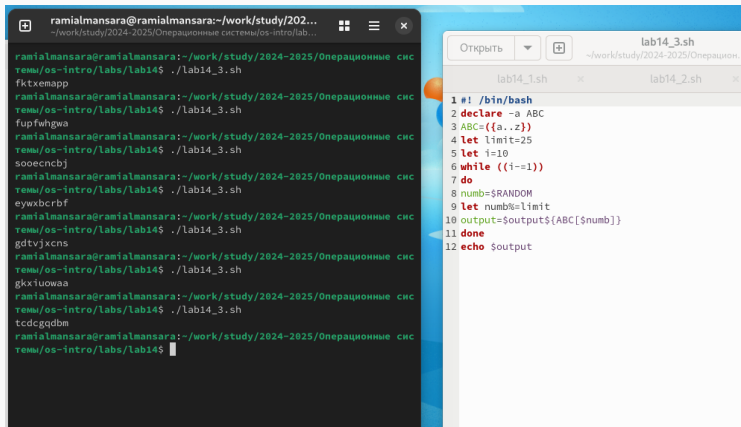


```
lab14_2.sh
~work/study/2024-2025/Операцион...
lab14_1.sh lab14_2.sh lab
1 #! /bin/bash
2 less /usr/share/man/man1/$1.1.gz
sh Ширина табуляции: 8
```

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 script editor on the right. The terminal window displays the execution of a script named `lab14_3.sh` in a directory `~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$`. The script's output consists of a series of random alphanumeric strings: `fktxemapp`, `fupfwhgwa`, `sooecncbj`, `eywxbcrbf`, `gdtvjxcns`, `gkx1uowaa`, and `tcdcqqdbm`. The script editor on the right shows the content of `lab14_3.sh`, which is a Bash script that declares a variable `ABC`, sets a `limit` of 25, and uses a `while` loop to generate random strings and store them in an array `output`.

```
ramialmansara@ramialmansara:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
fktxemapp
ramialmansara@ramialmansara:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
fupfwhgwa
ramialmansara@ramialmansara:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
sooecncbj
ramialmansara@ramialmansara:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
eywxbcrbf
ramialmansara@ramialmansara:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
gdtvjxcns
ramialmansara@ramialmansara:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
gkx1uowaa
ramialmansara@ramialmansara:~/work/study/2024-2025/Операционные системы/os-intro/labs/lab14$ ./lab14_3.sh
tcdcqqdbm
ramialmansara@ramialmansara:~/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   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.