

Программирование в командном процессоре ОС UNIX.

Ларина Наталья¹

24 апреля, 2024, Москва, Россия

¹Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

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

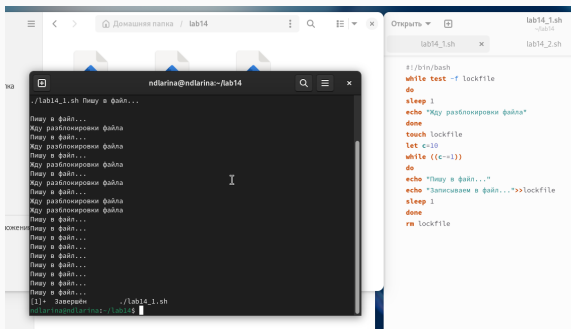
Задачи лабораторной работы

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 screenshot displays a Linux desktop environment. In the background, a file manager window shows the 'Домашняя папка' (Home folder) containing a directory named 'lab14'. In the foreground, a terminal window titled 'ndlarina@ndlarina:~/lab14' is open. The terminal shows the execution of a script named 'lab14_1.sh'. The script's output consists of a loop where the command 'echo "Пишу в файл..."' is printed multiple times, followed by 'Куда разблокировки файла' and another 'echo "Пишу в файл..."'. The prompt '[1]+ Завершен ./lab14_1.sh' indicates the script has finished. To the right of the terminal, a separate window displays the source code of 'lab14_1.sh'. The script is a Bash script that checks for the existence of a 'lockfile'. If it exists, it sleeps for 1 second, echoes a message, and touches the file. It then sets a counter 'c' to 10 and enters a 'while' loop that continues as long as 'c' is greater than 0. Inside this loop, it echoes the message, echoes a command to write to the file, sleeps for 1 second, and decrements 'c'. Finally, it removes the 'lockfile'.

```
#!/bin/bash
while test -f lockfile
do
sleep 1
echo "Куда разблокировки файла"
done
touch lockfile
let c=10
while ((c>0))
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

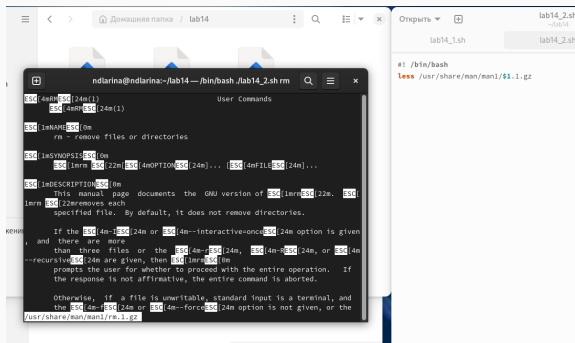
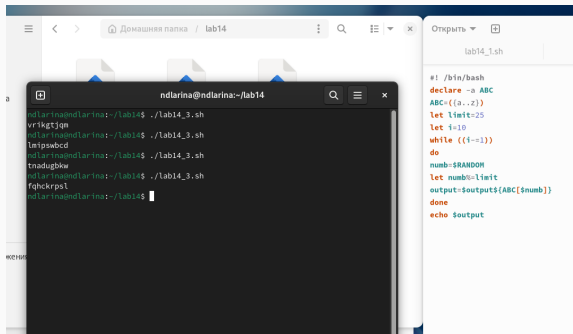


Рис. 2: Задание 2

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

Выполнение работы



The image shows a terminal window and a file editor. The terminal window, titled 'ndlarina@ndlarina:~/lab14', displays the execution of a script 'lab14_3.sh' multiple times, showing various random strings. The file editor, titled 'lab14_1.sh', shows the content of the script, which is a bash script that generates random strings of length 25.

```
ndlarina@ndlarina:~/lab14$ ./lab14_3.sh
vrikgjtjge
ndlarina@ndlarina:~/lab14$ ./lab14_3.sh
lmipswbcd
ndlarina@ndlarina:~/lab14$ ./lab14_3.sh
tnadugbkw
ndlarina@ndlarina:~/lab14$ ./lab14_3.sh
fqhckrpsl
ndlarina@ndlarina:~/lab14$
```

```
#!/bin/bash
declare -a ABC
ABC=(a..z)
let limit=25
let i=10
while ((i--))
do
    numb=$RANDOM
    let numb%=limit
    output=${output}${ABC[numb]}
done
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.