

Лабораторная работа №14

Презентация

Устинова В. В.

13 мая 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Устинова Виктория Вадимовна
- студент НПИбд-01-24
- Российский университет дружбы народов

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

1. Написать командный файл, реализующий упрощённый механизм семафоров. Командный файл должен в течение некоторого времени t_1 дожидаться освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использовать его в течение некоторого времени $t_2 < t_1$, также выдавая информацию о том, что ресурс используется соответствующим командным файлом (процессом).
2. Реализовать команду `man` с помощью командного файла. Изучите содержимое каталога `/usr/share/man/man1`. В нем находятся архивы текстовых файлов, содержащих справку по большинству установленных в системе программ и команд. Каждый архив можно открыть командой `less` сразу же просмотрев содержимое справки.
3. Используя встроенную переменную `$RANDOM`, напишите командный файл, генерирующий случайную последовательность букв латинского алфавита. Учтите, что `$RANDOM` выдаёт псевдослучайные числа в диапазоне от 0 до 32767.

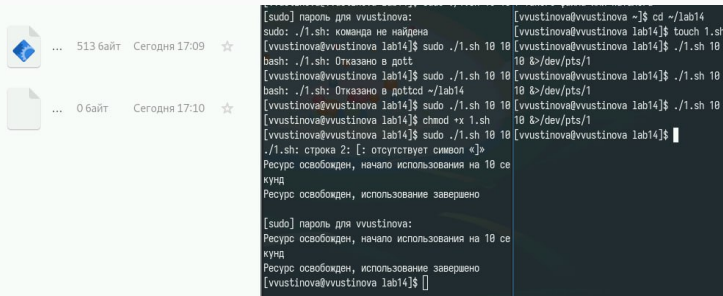
Первое задание

Написать командный файл, реализующий упрощённый механизм семафоров, командный файл должен в течение некоторого времени t_1 дожидаться освобождения ресурса, а дождавшись его освобождения, использовать его в течение некоторого времени $t_2 < t_1$

```
#!/bin/bash
if [ $# -ne 2 ]; then
    echo "Использование: $0 <t1> <t2>"
    exit 1
fi
t1=$1
t2=$2
semaphore_file="semaphore.lock"
touch $semaphore_file
function access_resource {
    while ! ln $semaphore_file $0.lock 2>/dev/null; do
        echo "Ресурс занят"
        sleep $t1
    done
    echo "Ресурс освобожден, начало использования на $t2 секунд"
    sleep $t2
    echo "Ресурс освобожден, использование завершено"
    rm $0.lock
}
access_resource
```

Первое задание

Запустить командный файл в одном виртуальном терминале в фоновом режиме, перенаправив его вывод в другой, в котором также запущен этот файл, но не фоновом, а в привилегированном режиме



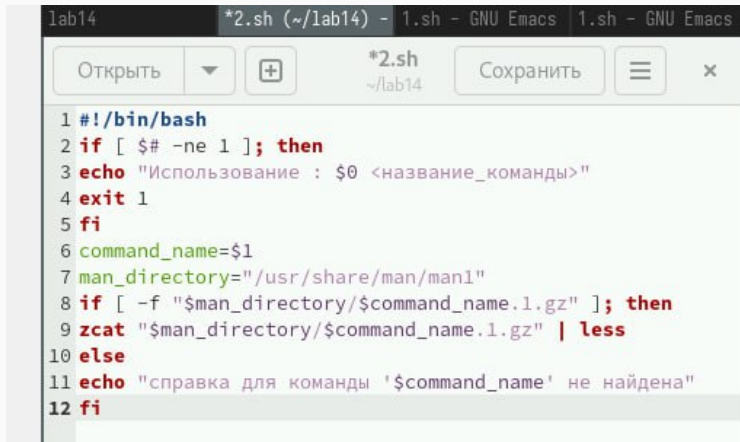
```
[sudo] пароль для vvustinova:
sudo: ./1.sh: команда не найдена
[vvustinova@vvustinova lab14]$ sudo ./1.sh 10 10
bash: ./1.sh: Отказано в dott
[vvustinova@vvustinova lab14]$ sudo ./1.sh 10 10
bash: ./1.sh: Отказано в dottcd ~/lab14
[vvustinova@vvustinova lab14]$ sudo ./1.sh 10 10
[vvustinova@vvustinova lab14]$ chmod +x 1.sh
[vvustinova@vvustinova lab14]$ sudo ./1.sh 10 10
./1.sh: строка 2: [: отсутствует символ «]»
Ресурс освобожден, начало использования на 10 се
кунд
Ресурс освобожден, использование завершено

[sudo] пароль для vvustinova:
Ресурс освобожден, начало использования на 10 се
кунд
Ресурс освобожден, использование завершено
[vvustinova@vvustinova lab14]$
```

Рис. 2: Ресурс освобожден, начало использования на 10сек

Второе задание

Реализовать команду `man` с помощью командного файла

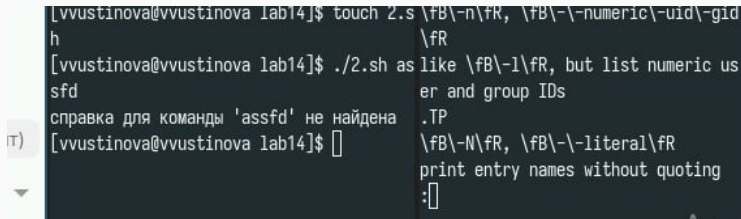


```
lab14 *2.sh (~/.lab14) - 1.sh - GNU Emacs 1.sh - GNU Emacs
Открыть *2.sh Сохранить x
1 #!/bin/bash
2 if [ $# -ne 1 ]; then
3 echo "Использование : $0 <название_команды>"
4 exit 1
5 fi
6 command_name=$1
7 man_directory="/usr/share/man/man1"
8 if [ -f "$man_directory/$command_name.1.gz" ]; then
9 zcat "$man_directory/$command_name.1.gz" | less
10 else
11 echo "справка для команды '$command_name' не найдена"
12 fi
```

Рис. 3: Каждый архив можно открыть командой `less` сразу же просмотрев содержимое справки

Второе задание

Выводим команду man



```
[vvustinova@vvustinova lab14]$ touch 2.s \fB\n\fR, \fB\--numeric\uid\gid
h
[vvustinova@vvustinova lab14]$ ./2.sh as like \fB\l\fR, but list numeric us
sfd er and group IDs
справка для команды 'assfd' не найдена .TP
[vvustinova@vvustinova lab14]$ \fB\N\fR, \fB\--literal\fR
print entry names without quoting
:

```

Рис. 4: Правильный вывод, слева не выводит,справа вывод команды ls

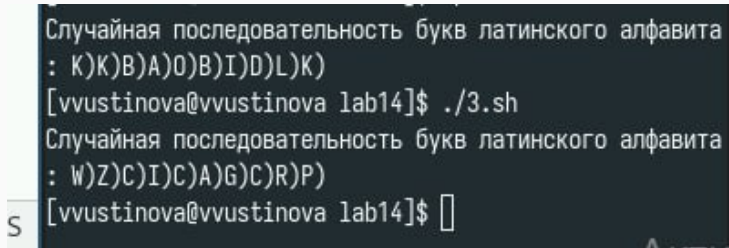
Третье задание

Используя встроенную переменную \$RANDOM, напишите командный файл, генерирующий случайную последовательность букв латинского алфавита

```
1 #!/bin/bash
2
3 generate_random_letter() {
4     # Случайное число от 0 до 25
5     random_number=$((RANDOM % 26))
6     letter=$(printf "\\$(printf '%03o' $((65 + random_number)))")
7     echo -n "$letter"
8 }
9
10 random_sequence=""
11 for ((i=0; i<10; i++)); do
12     random_sequence="$random_sequence$(generate_random_letter)"
13 done
14
15 echo "Случайная последовательность букв латинского алфавита:
16     $random_sequence"
17
```

Рис. 5: Пишем командный файл

Случайная последовательность

A terminal window with a dark background and light-colored text. The text shows a script being executed that prints two random sequences of uppercase letters. The first sequence is 'K)K)B)A)O)B)I)D)L)K)' and the second is 'W)Z)C)I)C)A)G)C)R)P)'. The prompt is '[vvustinova@vvustinova lab14]\$'.

```
Случайная последовательность букв латинского алфавита  
: K)K)B)A)O)B)I)D)L)K)  
[vvustinova@vvustinova lab14]$ ./3.sh  
Случайная последовательность букв латинского алфавита  
: W)Z)C)I)C)A)G)C)R)P)  
[vvustinova@vvustinova lab14]$
```

Рис. 6: Выводит рандомную псоледовательность

Мы успешно изучили основы программирования в оболочке ОС UNIX. Научились писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.