

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 12

дисциплина: Операционные системы

Студент: Давитян Эдуард

Группа: НПИ-01-21

Цель работы:

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

Задание:

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). Запустить командный файл в одном виртуальном терминале в фоновом режиме, перенаправив его вывод в другой ($> /dev/tty\#$, где $\#$ — номер терминала куда перенаправляется вывод), в котором также запущен этот файл, но не фоновом, а в привилегированном режиме. Доработать программу так, чтобы имела возможность взаимодействия трёх и более процессов
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. Командный файл должен получать в виде аргумента командной строки название команды и в виде результата выдавать справку об этой команде или сообщение об отсутствии справки, если соответствующего файла нет в каталоге `man1`.
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.

Выполнение работы:

1.script 1

```
eduard.davitiani@localhost:/tmp/lab12
Файл Правка Вид Поиск Терминал Справка
exec {fn}>"$lockfile"
if test -f "$lockfile"
then
    while [ 1!=0]
    do
        if flock -n ${fn}
        then
            echo "file was locked"
            sleep 4
            echo "unlocking"
            flock -u ${fn}

        else
            echo "file already locked"
            sleep 3
        fi
    done
fi
~
```

```
eduard.davitiani@localhost:/tmp/lab12
Файл Правка Вид Поиск Терминал Справка
[eduard.davitiani@localhost tmp]$ mkdir lab12
mkdir: невозможно создать каталог «lab12»: Файл существует
[eduard.davitiani@localhost tmp]$ cd lab12
[eduard.davitiani@localhost lab12]$ touch script1
[eduard.davitiani@localhost lab12]$ vi script 1
```

```
eduard.davitiani@localhost:/tmp/lab12
Файл Правка Вид Поиск Терминал Справка
[eduard.davitiani@localhost lab12]$ chmod 777 script1
[eduard.davitiani@localhost lab12]$ ./script1
./script1: строка 6: [: отсутствует символ «]»
[eduard.davitiani@localhost lab12]$ vi script1
[eduard.davitiani@localhost lab12]$ ./script1
./script1: строка 5: синтаксическая ошибка рядом с неожиданным маркером «do»
./script1: строка 5: `        do'
[eduard.davitiani@localhost lab12]$ vi script1
[eduard.davitiani@localhost lab12]$ ./script1
file was locked
unlocking
file was locked
unlocking
file was locked
unlocking
file was locked
unlocking
file was locked
```

2.script2

```
eduard.davitiani@localhost:~/lab12
Файл Правка Вид Поиск Терминал Справка
TOUCH(1) User Commands TOUCH(1)
ESC[1mNAMEESC[0m
touch - change file timestamps
ESC[1mSYNOPSISESC[0m
ESC[1mtouch ESC[22m[ESC[4mOPTIONESC[24m]... ESC[4mFILEESC[24m]...
ESC[1mDESCRIPTIONESC[0m
Update the access and modification times of each FILE to the current
time.
A FILE argument that does not exist is created empty, unless ESC[1m-c
ESC[22mor ESC[1m-hESC[0m
is supplied.
A FILE argument string of - is handled specially and causes touch to
change the times of the file associated with standard output.
Mandatory arguments to long options are mandatory for short options
too.
ESC[1m-a ESC[22mchange only the access time
/usr/share/man/man1/touch.1.gz
```

```
eduard.davitiani@localhost:/usr/share/man/man1
Файл Правка Вид Поиск Терминал Справка
[eduard.davitiani@localhost lab12]$ cd /usr/share/
[eduard.davitiani@localhost share]$ cd man
[eduard.davitiani@localhost man]$ cd man1
[eduard.davitiani@localhost man1]$ ls
```

```

bash: ./script: Нет такого файла или каталога
[eduard.davitiani@localhost lab12]$ ./script2 -n asd
./script2: строка 11: синтаксическая ошибка: неожиданный конец файла
[eduard.davitiani@localhost lab12]$ vi script2
[eduard.davitiani@localhost lab12]$ ./script2 -n asd
asd
[eduard.davitiani@localhost lab12]$

```

```

Файл  Правка  Вид  Поиск  Терминал  Справка
command=""

while getopts :n: opt
do
case $opt in
n)command="$OPTARG";;
esac
done
if test -f "/usr/share/man/man1/$command.1.gz"
then less /usr/share/man/man1/4COMMAND.1.gz
else
echo "no such command"
fi
~
~
~
~
~
~
~
~
~
~
:wq

```

3.script3

```

eduard.davitiani@localhost:~/lab12
Файл  Правка  Вид  Поиск  Терминал  Справка
[eduard.davitiani@localhost lab12]$ touch script3
[eduard.davitiani@localhost lab12]$ vi script3
[eduard.davitiani@localhost lab12]$ chmod 777 script3
[eduard.davitiani@localhost lab12]$ ./script3
fagi
[eduard.davitiani@localhost lab12]$ ./script3
bbbbbd
[eduard.davitiani@localhost lab12]$

```

```

Файл  Правка  Вид  Поиск  Терминал  Справка
echo $RANDOM | tr '0-9' 'a-zA-Z'
~
~
~
~
~

```

Выводы:

В ходе данной лабораторной работы изучили основы программирования в оболочке ОС UNIX

Контрольные вопросы:

- Так же между скобками должны быть пробелы
- `Cat file.txt | xargs | sed -e 's/\\. / . \\n/g'`
- Seq- выдает последовательность чисел. Реализовать ее функционал можно командой `for n in`
- 3
- Zsh очень сильно упрощает работу. Но существует различия . Например , в zsh после `for` обязательно вставлять пробел , нумерация массивов в zsh начинается с 1
- верен
- Bash позволяет очень легко работать с файловой системой без лишних конструкций . Но относительно обычных языков программирования bash очень сжат