

# Лабораторная работа №12. Программирование в командном процессоре ОС UNIX. Расширенное программирование

---

Парфенова Елизавета Евгеньевна

RUDN University, Moscow, Russian Federation

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). 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 имелась 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.

# Первый командный файл

Начинаем с первого задания. Вначале создаем файл script1 и записываем туда код командного файла. Осуществляем его с помощью циклов if и while. (рис. 1)

```
lockfile="./locking.file"
exec {fn}>$lockfile
if test -f "$lockfile"
then
    while [ 1!=0 ]
    do
        if flock -n ${fn}
        then
            echo "File was locked"
            sleep 3
            echo "Unlocking"
            flock -u ${fn}
        else
            echo "File already locked"
            sleep 3
        fi
    done
fi
```

Figure 1: Код первого командного файла



Приступаем к созданию второго скрипта. В нем нужно было реализовать команду man. Делаем это с помощью getoptс и разных циклов. (рис. 3)

```
command=""

while getoptс :m: opt
do
    case $opt in
        m) command="$OPTARG";;
        esac
    done

    if test -f "/usr/share/man/man1/$command.1.gz"
    then less "/usr/share/man/man1/$command.1.gz"
    fi
```

Figure 3: Код второго командного файла

Затем делаем файл исполняемым и запускаем его командой `**./script2 -m ls`, посмотрев справку команды `ls`. Справка успешно открылась. (рис. 4) (рис. 5)

```
[eeparfenova@fedora lab12]$ chmod +x script2
[eeparfenova@fedora lab12]$ ./script2 -m ls
```

Figure 4: Запуск второго командного файла



Figure 5: Работа второго командного файла (справка по команде)



Приступаем к написанию последнего командного файла. Создаем файл `script3` и записываем туда код. Вначале выводим на экран нужную надпись а после, используя встроенную переменную, генерируем одну последовательность букв. (рис. 6)

```
echo "Random letter sequence"  
cat /dev/urandom | tr -dc "a-zA-Z0-9" | fold -w 15 | head -n 1
```

**Figure 6:** Код третьего командного файла

## Третий командный файл

После делаем файл исполняемым и несколько раз вызываем его командой `./script3`. Скрипт сработал успешно. (рис. 7)

```
[eeparfenova@fedora lab12]$ chmod +x script3
[eeparfenova@fedora lab12]$ ./script3
Random letter sequence
ajaa87aWs5VR8kd
[eeparfenova@fedora lab12]$ ./script3
Random letter sequence
qHpKbK2DgYRwjGo
[eeparfenova@fedora lab12]$ ./script3
Random letter sequence
uxvorDzFKwuxFFS
[eeparfenova@fedora lab12]$ ./script3
Random letter sequence
fF3wzMAZvydd7j4
[eeparfenova@fedora lab12]$ ./script3
Random letter sequence
jTioD9pS6ju4t0l
```

Figure 7: Работа третьего командного файла

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