

## Лабораторная работа 12

---

Генералов Даниил, НПИ-01-21, 10322122800

28 May, 2022

<sup>1</sup>RUDN University, Moscow, Russian Federation

## Цель и задачи

---

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

Требуется написать 3 командных файла:

- берет эксклюзивное управление файлом и держит его некоторое время
- реализует отображение **man**-страниц
- выводит последовательность случайных символов

## Выполнение работы

---

# Программа для блокирования файлов

```
8 > lab12 > code > 1-flock.sh
1  #!/bin/bash
2
3  if [[ -z "$1" || -z "$2" || -z "$3" ]]; then
4      echo "Usage: $0 <file> <timeout> <lock-time> [id]"
5      exit 1
6  fi
7
8  if [[ ! -f "$1" ]]; then
9      echo "$4: File $1 does not exist, creating it"
10     touch "$1"
11 fi
12
13 {
14     echo "$4 Waiting for lock on $1 for $2 seconds"
15     flock --verbose -x -w "$2" 3 || { echo "$4 Lock acquisition failed after $2 seconds" && exit 1; }
16     echo "$4 Acquired a lock, holding it for $3..."
17     sleep $3
18     echo "$4 Releasing lock"
19 } 3>> "$1"
```

Figure 1: Программа для блокирования файлов

# Программа для отображения man-страниц

```
1  #!/bin/bash
2
3  if [[ -z "$1" ]]; then
4      echo "Usage: $0 <man-page>"
5      exit 1
6  fi
7
8  if [[ ! -f "/usr/share/man/man1/$1.1.gz" ]]; then
9      echo "No manual page for $1 found in section 1"
10     exit 1
11 fi
12
13 zcat "/usr/share/man/man1/$1.1.gz" | groff -T utf8 -man | less -R
```

Figure 2: Программа для отображения man-страниц

## Программа для вывода случайных символов

```
labs > lab12 > code > 3-random.sh
1  #!/bin/bash
2
3  while true; do
4      charind=$((RANDOM % 26 + 97))
5      charcode=`printf '%x' $charind`
6      char=`printf "\x$charcode"`
7      echo -n $char
8      sleep 0.1
9  done
```

Figure 3: Программа для вывода случайных символов



## Заключение

---

Командные файлы – это удобный способ писать программы в той же самой среде, в которой работает пользователь.

Мы закончили знакомство со средствами разработки командных файлов, рассмотрев более сложные примеры.