

# Front matter

lang: ru-RU title: "Лабораторная работа №14" subtitle: " Программирование в командном процессоре ОС UNIX. Расширенное программирование" author: "Данилова Анастасия Сергеевна"

## Formatting

toc-title: "Содержание" toc: true # Table of contents toc\_depth: 2 lof: true # List of figures lot: true # List of tables  
fontsize: 12pt linestretch: 1.5 papersize: a4paper documentclass: scrreprt polyglossia-lang: russian polyglossia-  
otherlangs: english mainfont: PT Serif romanfont: PT Serif sansfont: PT Sans monofont: PT Mono mainfontoptions:  
Ligatures=TeX romanfontoptions: Ligatures=TeX sansfontoptions: Ligatures=TeX,Scale=MatchLowercase  
monofontoptions: Scale=MatchLowercase indent: true pdf-engine: lualatex header-includes:

- `\linepenalty=10` # the penalty added to the badness of each line within a paragraph (no associated penalty node) Increasing the value makes tex try to have fewer lines in the paragraph.
- `\interlinepenalty=0` # value of the penalty (node) added after each line of a paragraph.
- `\hyphenpenalty=50` # the penalty for line breaking at an automatically inserted hyphen
- `\exhyphenpenalty=50` # the penalty for line breaking at an explicit hyphen
- `\binoppenalty=700` # the penalty for breaking a line at a binary operator
- `\relpenalty=500` # the penalty for breaking a line at a relation
- `\clubpenalty=150` # extra penalty for breaking after first line of a paragraph
- `\widowpenalty=150` # extra penalty for breaking before last line of a paragraph
- `\displaywidowpenalty=50` # extra penalty for breaking before last line before a display math
- `\brokenpenalty=100` # extra penalty for page breaking after a hyphenated line
- `\prelispdisplaypenalty=10000` # penalty for breaking before a display
- `\postdisplaypenalty=0` # penalty for breaking after a display
- `\floatingpenalty = 20000` # penalty for splitting an insertion (can only be split footnote in standard LaTeX)
- `\raggedbottom` # or `\flushbottom`
- `\usepackage{float}` # keep figures where there are in the text
- `\floatplacement{figure}{H}` # keep figures where there are in the text

---

## 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 `t1` дожидаться освобождения ресурса, выдавая об этом сообщение, а дождавшись его освобождения, использовать его в течение некоторого времени `t2 <> t1`, 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.
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.

# Теоретическое введение

Less. Утилиту less зачастую используют не для чтения текста, а для поиска определенных участков в больших документах. Команда less в Linux пригодится для просмотра по-настоящему больших текстовых файлов, с которыми затруднительно работать в текстовых редакторах вроде vim или с помощью утилит, загружающих весь документ сразу.

\$RANDOM - внутренняя функция Bash (не константа), которая возвращает *псевдослучайные* целые числа в диапазоне 0 - 32767. Функция \$RANDOM не должна использоваться для генерации ключей шифрования.

## Выполнение лабораторной работы

1. Реализуем команду man с помощью командного файла. Изучим 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. (рис 1)



```
#!/bin/bash
cd /usr/share/man/man1
less $1*
```

рис 1 Введем команду chmod ugo+x filename, для разрешения исполнения файла для всех. (рис 1.2)



```
anastasia@anastasia-VirtualBox:~$ touch lab13
anastasia@anastasia-VirtualBox:~$ nano lab13
anastasia@anastasia-VirtualBox:~$ chmod ugo+x lab13
anastasia@anastasia-VirtualBox:~$ ./lab13
```

рис 1.2 Далее убедимся, что программа работает (рис 2)

```
.\" DO NOT MODIFY THIS FILE! It was generated by help2man 1.47.3.
.TH TEST "1" "September 2019" "GNU coreutils 8.30" "User Commands"
.SH NAME
test \- check file types and compare values
.SH SYNOPSIS
.B test
.I EXPRESSION
.br
.B test
.br
.\" \& tells docclifter the brackets are literal (Bug#31803).
.B [\&
.I EXPRESSION
.B ]\&
.br
.B "[\& ]\&"
.br
.B [\&
.I OPTION
.SH DESCRIPTION
.\" Add any additional description here
.PP
Exit with the status determined by EXPRESSION.
[.1.gz (file 1 of 1749)]
```

рис 2

- Используя встроенную переменную \$RANDOM, напомним командный файл, генерирующий случайную последовательность букв латинского алфавита. Учтем, что \$RANDOM выдаёт псевдослучайные числа в диапазоне от 0 до 32767, поэтому поставим ограничение. (рис 3)

```
GNU nano 4.8 lab13 Изменён
#!/bin/bash
echo $RANDOM | tr '[0-9]' '[a-z]'
```

рис 3 Убедимся, что программа действительно выводит рандомный набор латинских букв (рис 4)

```
anastasia@anastasia-VirtualBox:~$ nano lab13
anastasia@anastasia-VirtualBox:~$ chmod ugo+x lab13
anastasia@anastasia-VirtualBox:~$ ./lab13
cedfh
anastasia@anastasia-VirtualBox:~$ ./lab13
ctiga
anastasia@anastasia-VirtualBox:~$ ./lab13
dbgef
anastasia@anastasia-VirtualBox:~$ ./lab13
hbaj
anastasia@anastasia-VirtualBox:~$
```

рис 4

## Выводы

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

## Библиография

[https://www.opennet.ru/docs/RUS/bash\\_scripting\\_guide/x4812.html](https://www.opennet.ru/docs/RUS/bash_scripting_guide/x4812.html)

<https://losst.ru/komanda-less-v-linux>