
Front matter

lang: ru-RU title: Программирование в UNIX 3: Месть программиста subtitle: ДР по ОС №13 author: Аникин Константин Сергеевич group: НПИбд-01-20

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 - \predisplaypenalty=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

Цели работы

- Изучить основы программирования в оболочке ОС UNIX/Linux.
- Научиться писать более сложные командные файлы.

Задание

- Ознакомиться с теоретическим материалом.
- Написать три скрипта из задания.

Теоретическое введение

Для написания скрипта 1 необходимо выяснить, что такое семафоры. [\[1\]](#)

Семафор – это механизм, который позволяет конкурирующим процессам и потокам работать с общими ресурсами и помогает в решении различных проблем синхронизации таких как гонки, дедлоки (взаимные блокировки) и неправильное поведение потоков.

Для решения этих проблем в ядре присутствуют такие средства как мьютексы, семафоры, сигналы и барьеры.

Есть три вида семафоров:

- Бинарные семафоры (binary semaphore)
- Семафоры-счетчики (counting semaphore)
- Массивы семафоров (semaphore set)

Краткий справочник команд Linux [\[2\]](#)

Выполнение лабораторной работы

Скрипт 1

Непосредственно сам скрипт. (рис. 1)

Пока файл открыт, проверяем его на использование. Если файл используется, выводим Locked,

ждём 3 секунды и перестаём использовать файл. Если файл уже используется, выводим Couldn't lock,

ждём 4 секунды и пробуем открыть файл снова.

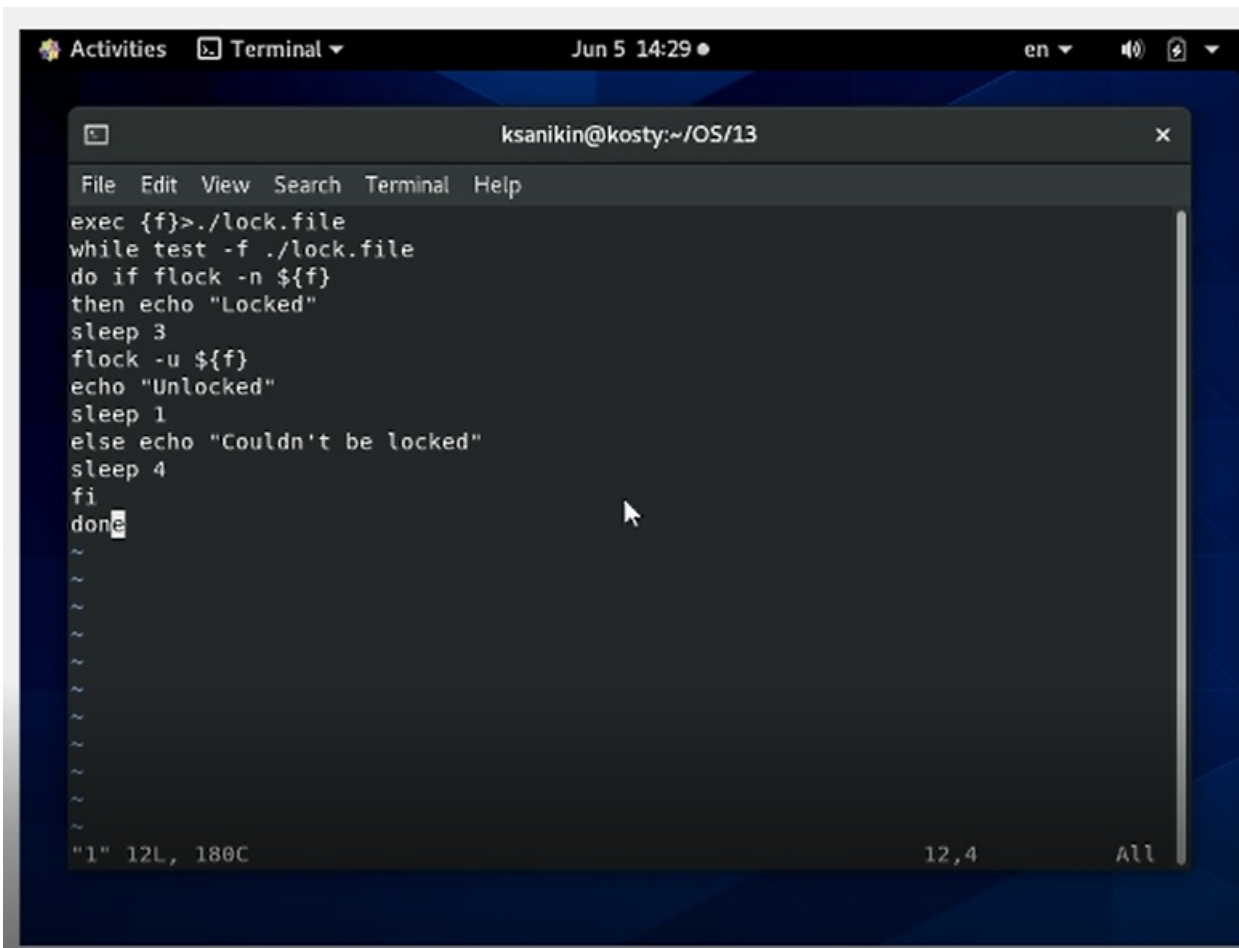


Рис. 1: Скрипт 1

Результаты работы скрипта 1, первый терминал (рис. 2)

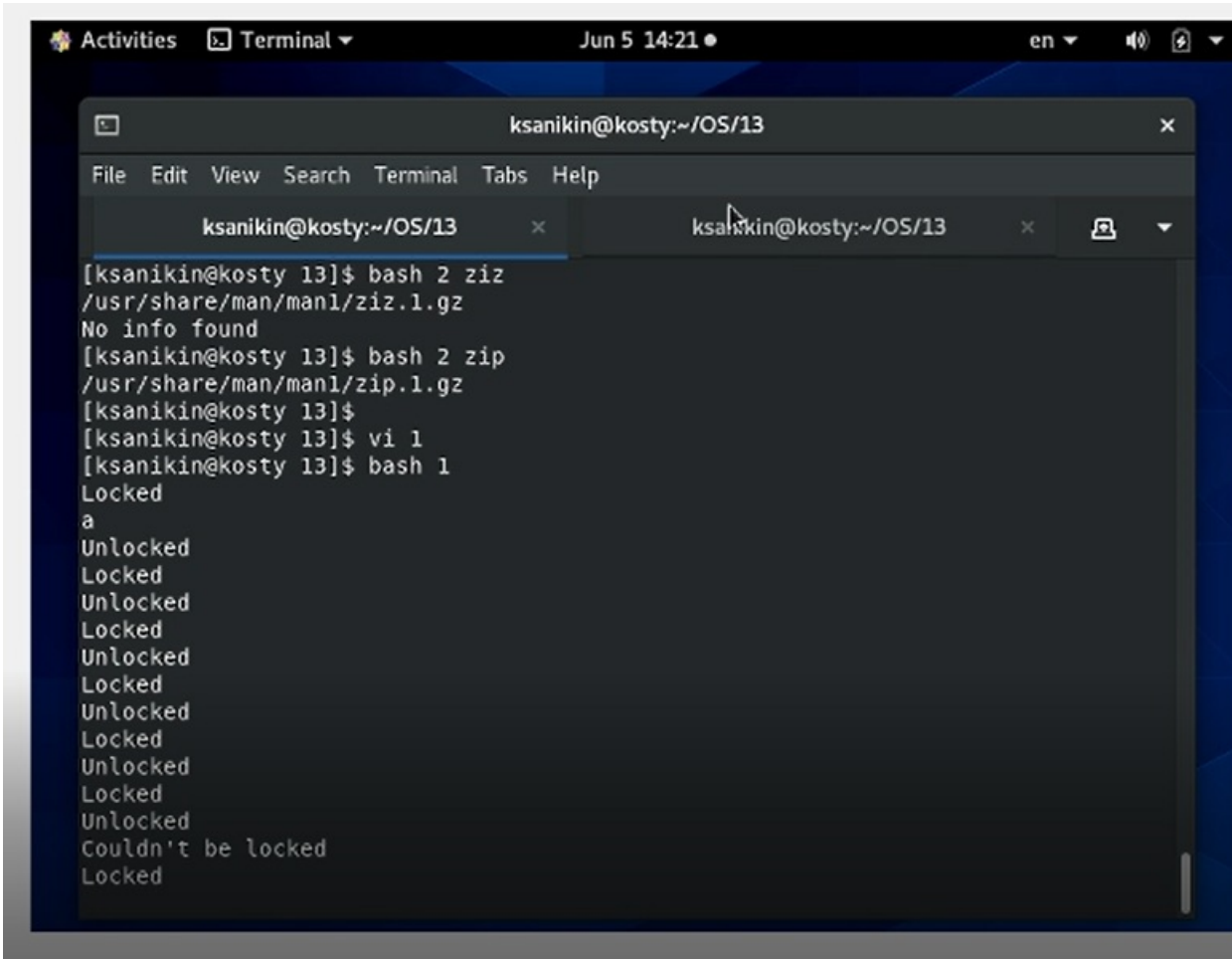


Рис. 2: Результаты работы скрипта 1

Результаты работы скрипта 1, второй терминал (рис. 3)

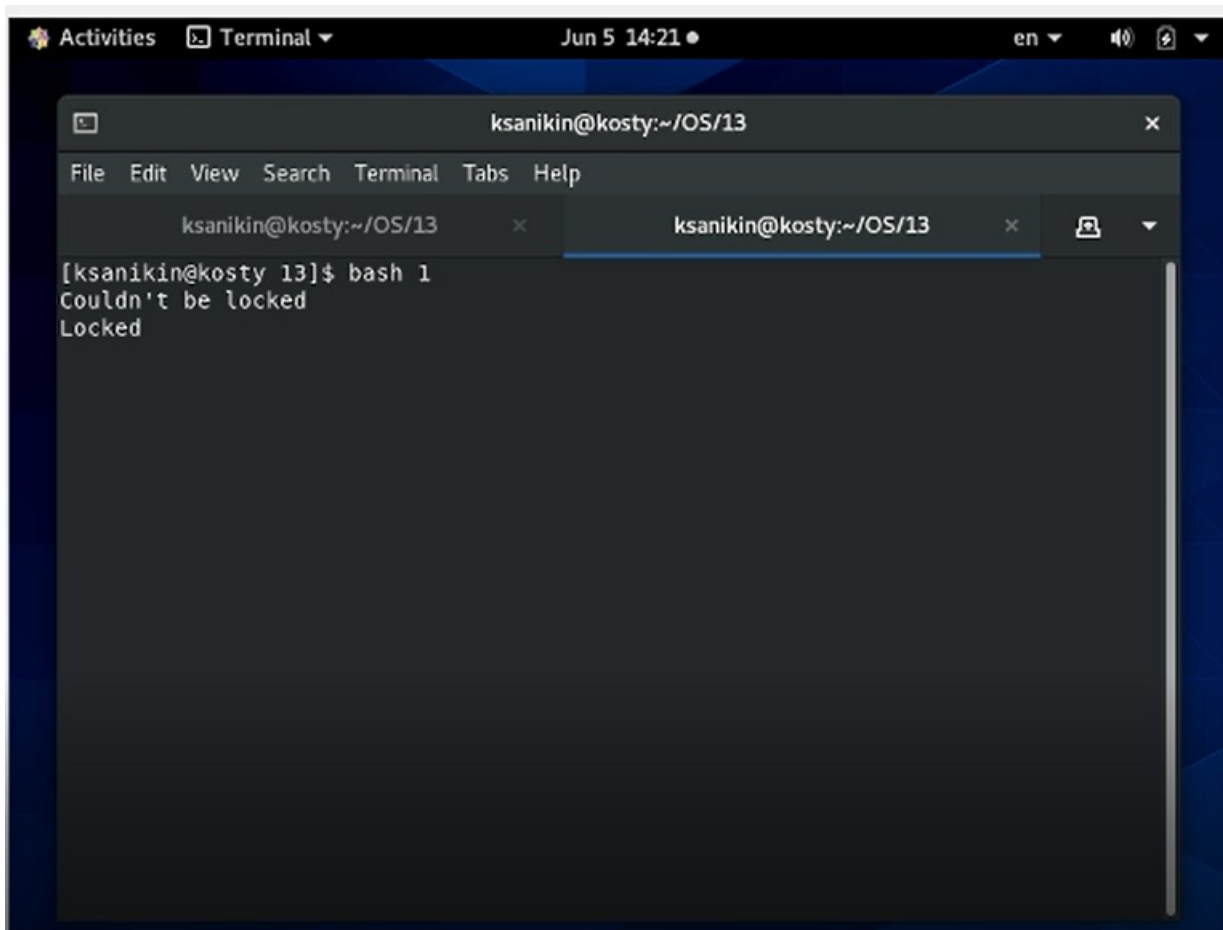


Рис. 3: Результаты работы скрипта 1

Скрипт 2

Непосредственно сам скрипт. (рис. 4)

Определяем строчку, переданную аргументом. Проверяем, существует ли файл.

Если да, командой less выводим справку о нём. Иначе выводим "No info found".

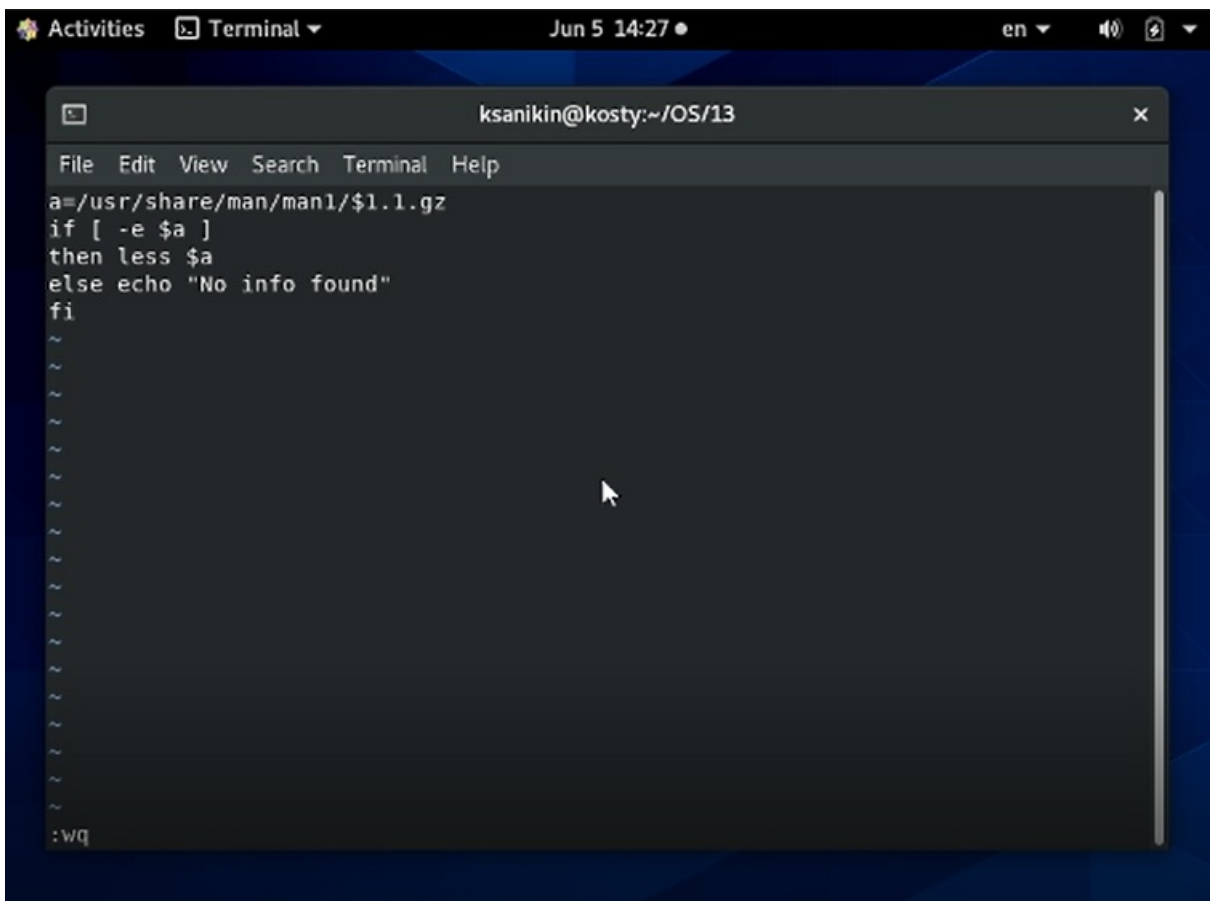


Рис. 4: Скрипт 2

Результаты работы скрипта 2 с несуществующей справкой (рис. 5)

(выведение пути к файлу - остаток от дебага кода, в финальной версии путь не выводится)

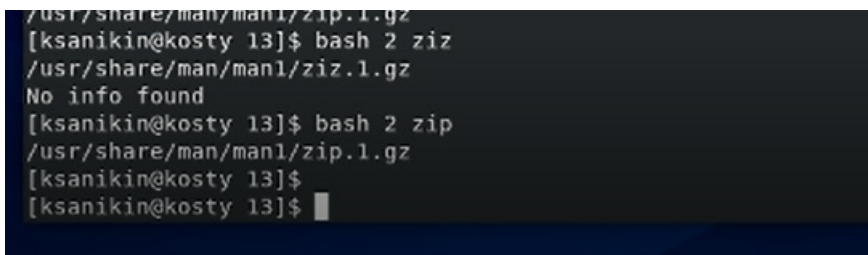


Рис. 5: Результаты работы скрипта 2

Результаты работы скрипта 2 с существующей справкой (рис. 6)

```
ksanikin@kosty:~/OS/13
File Edit View Search Terminal Help
ZIP(1L) ZIP(1L)
ESC[1mNAMEESC[0m
zip - package and compress (archive) files
ESC[1mSYNOPSISESC[0m
ESC[1mzip ESC[22m[-ESC[1maABcdDeEfFghjklLmoqrRSTuvVwXyz!@$ESC[22m] [--lo
ngoption ...] [-ESC[1mb ESC[22mpath]
[-ESC[1mn ESC[22msuffixes] [-ESC[1mt ESC[22mdate] [-ESC[1mtt ESC[22mdate]
[ESC[4mzipfileESC[24m [ESC[4mfileESC[24m ...]] [ESC[1m-xi ESC[22mlist]
ESC[1mzipcloak ESC[22m(see separate man page)
ESC[1mzipnote ESC[22m(see separate man page)
ESC[1mzipsplit ESC[22m(see separate man page)
Note: Command line processing in ESC[4mzipESC[24m has been changed to su
pport long
options and handle all options and arguments more consistently. Some
old command lines that depend on command line inconsistencies may no
longer work.
/usr/share/man/man1/zip.1.gz
```

Рис. 6: Результаты работы скрипта 2

Скрипт 3

Непосредственно сам скрипт. (рис. 7)

Создаём массив символов, после чего в цикле указанное аргументом число раз
выбираем случайный элемент массива и выводим его на экран

