

# Лабораторная работа 9

## Отчет по лабораторной работе 9

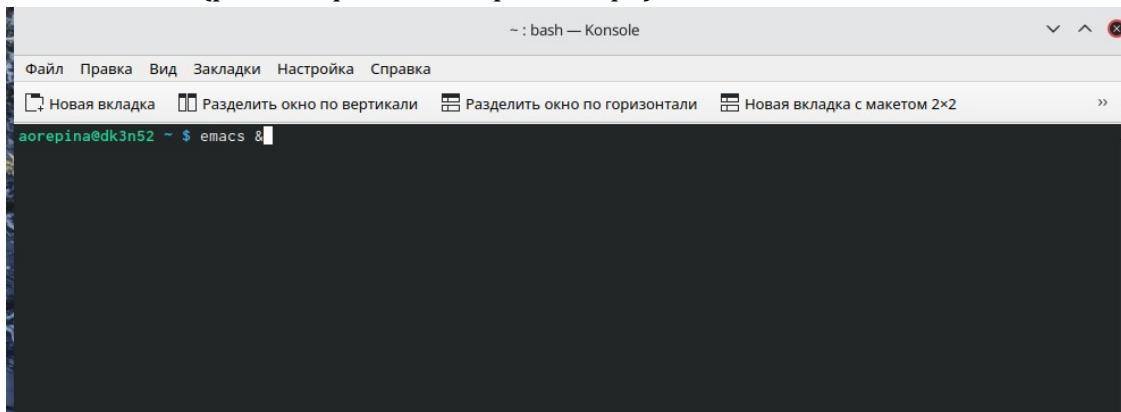
Репина Ангелина Олеговна

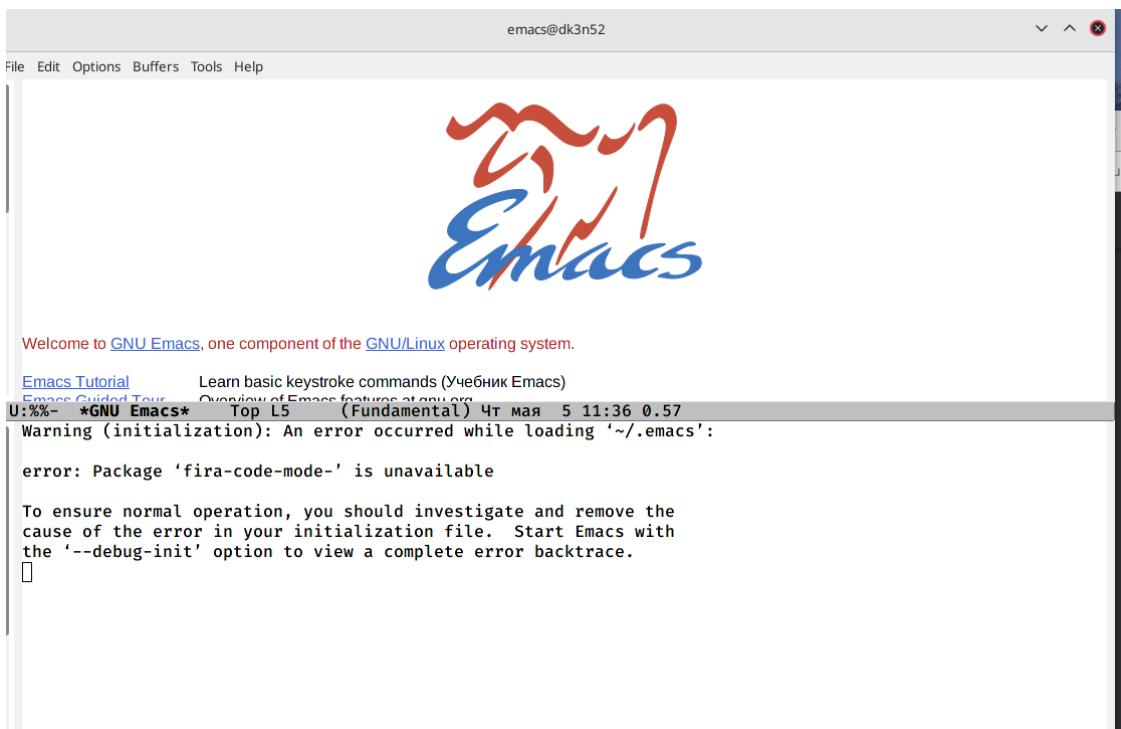
### Цель работы

Цель данной лабораторной работы - ознакомление с операционной системой Linux и получение практических навыков работы с редактором Emacs

### Выполнение лабораторной работы

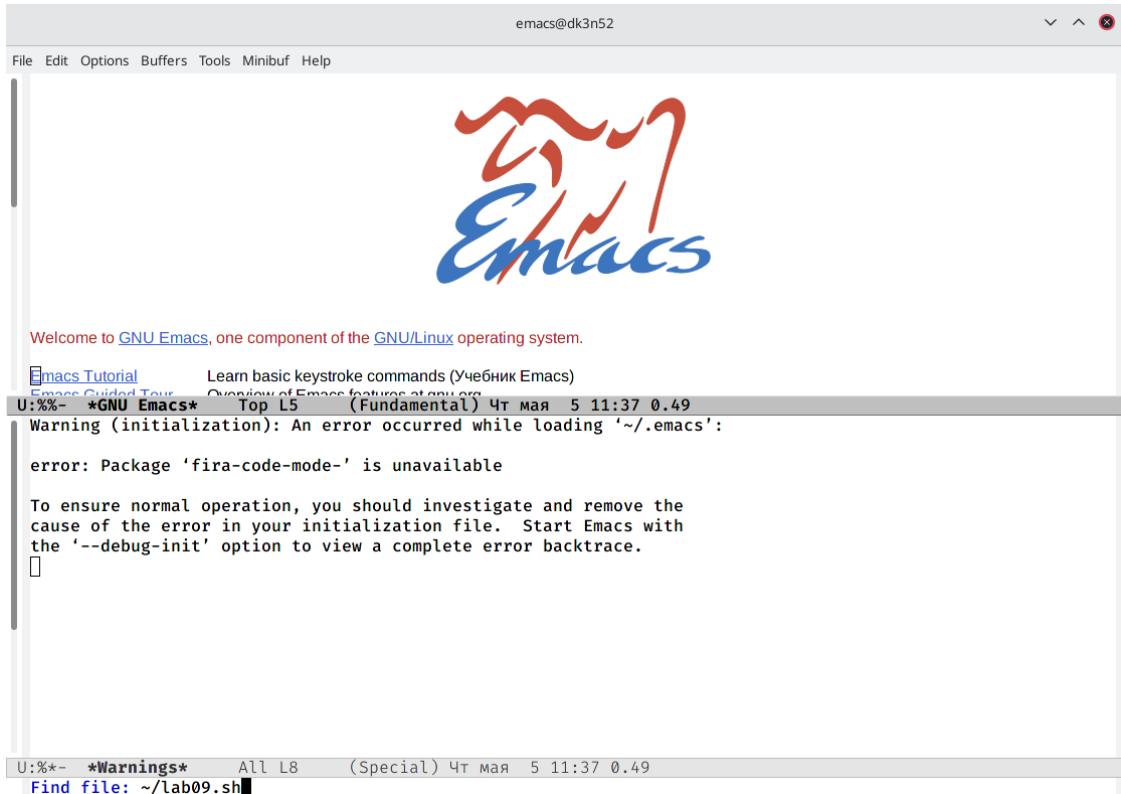
1. Ознакомилась с теоретическим материалом
2. Ознакомилась с редактором Emacs
3. Приступила к выполнению упражнения 1)Открыла emacs с помощью команды emacs & (рис 1, 2, рис 2 - вид редактора)





2

2) Создала файл lab09.sh с помощью комбинации клавиш Ctrl-x Ctrl-f (рис 3)



3

- 3) Набрала текст из лабораторной работы (рис 4)

The screenshot shows an Emacs window with a shell buffer. The buffer contains a shell script named 'lab09.sh' with the following content:

```
#!/bin/bash
HELL-Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
```

Below the script, there is an error message from Emacs:

```
U:**- lab09.sh All L8 (Shell-script[sh]) Чт мая 5 11:40 0.36
Warning (initialization): An error occurred while loading '~/.emacs':
error: Package 'fira-code-mode-' is unavailable

To ensure normal operation, you should investigate and remove the
cause of the error in your initialization file. Start Emacs with
the '--debug-init' option to view a complete error backtrace.
```

At the bottom of the window, there is a buffer titled '\*Warnings\*' containing the same error message:

```
U:%*- *Warnings* All L8 (Special) Чт мая 5 11:40 0.36
error: Package 'fira-code-mode-' is unavailable
```

4

- 4) Сохранила файл с помощью комбинации клавиш Ctrl-x Ctrl-s
- 5) Проделала стандартные процедуры редактирования, каждое действие осуществляла комбинацией клавиш а) Вырезала одой командой целую строку (Ctrl-k) (рис 5)

The screenshot shows an Emacs window with two buffers. The top buffer, titled 'lab09.sh', contains a shell script:

```
#!/bin/bash
HELL-Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
hello
```

The bottom buffer, titled '\*Warnings\*', displays a warning message:

```
U:***- lab09.sh      All L7      (Shell-script[sh]) Чт май 5 11:41 0.40
Warning (initialization): An error occurred while loading '~/.emacs':
error: Package 'fira-code-mode-' is unavailable

To ensure normal operation, you should investigate and remove the
cause of the error in your initialization file. Start Emacs with
the '--debug-init' option to view a complete error backtrace.
```

5

6) Вставила эту строку в конец файла (Ctrl-y) (рис 6)

The screenshot shows an Emacs window with two buffers. The top buffer, titled 'lab09.sh', contains the same shell script as before, but with an additional line at the end:

```
#!/bin/bash
HELL-Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
hello
echo $HELLO
```

The bottom buffer, titled '\*Warnings\*', displays the same warning message as in the previous screenshot:

```
U:***- lab09.sh      All L9      (Shell-script[sh]) Чт май 5 11:42 0.48
Warning (initialization): An error occurred while loading '~/.emacs':
error: Package 'fira-code-mode-' is unavailable

To ensure normal operation, you should investigate and remove the
cause of the error in your initialization file. Start Emacs with
the '--debug-init' option to view a complete error backtrace.
```

6

в) Выделила область текста (Ctrl-Space), скопировала её в буфер обмена (Alt-w), вставила область в конец файла (Ctrl-y) (рис 7)

```
emacs@dk3n52

File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

hello
echo $HELLO
function hello {
```

7

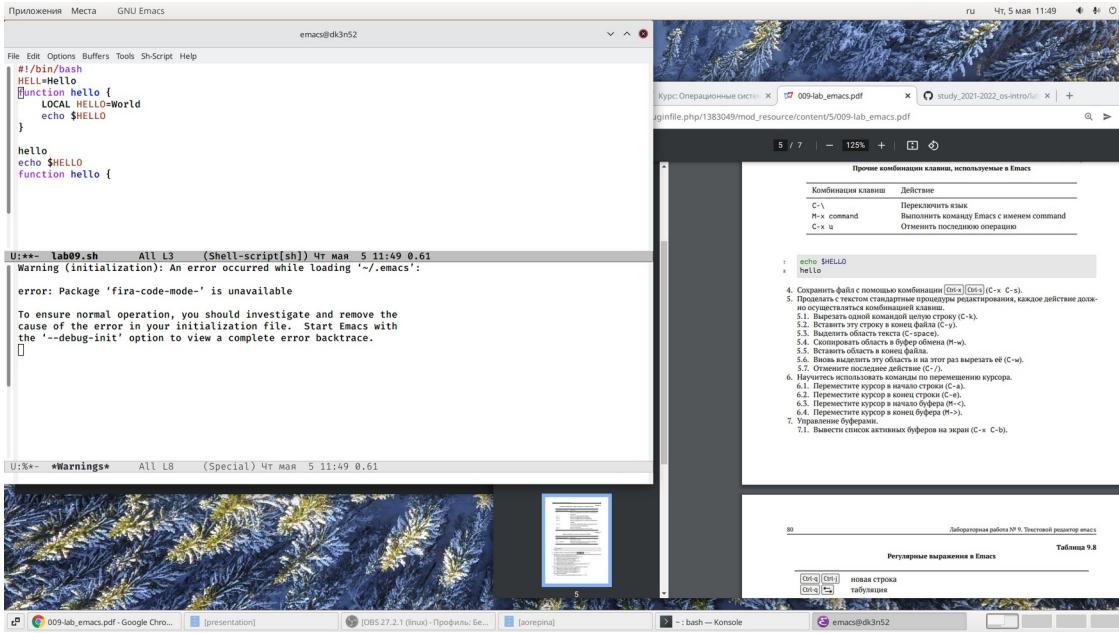
г) Вновь выделила эту область (Ctrl-Space) и на этот раз вырезала её (Ctrl-w)

д) Отменила последнее действие (Ctrl-/, u) (рис 8)

8

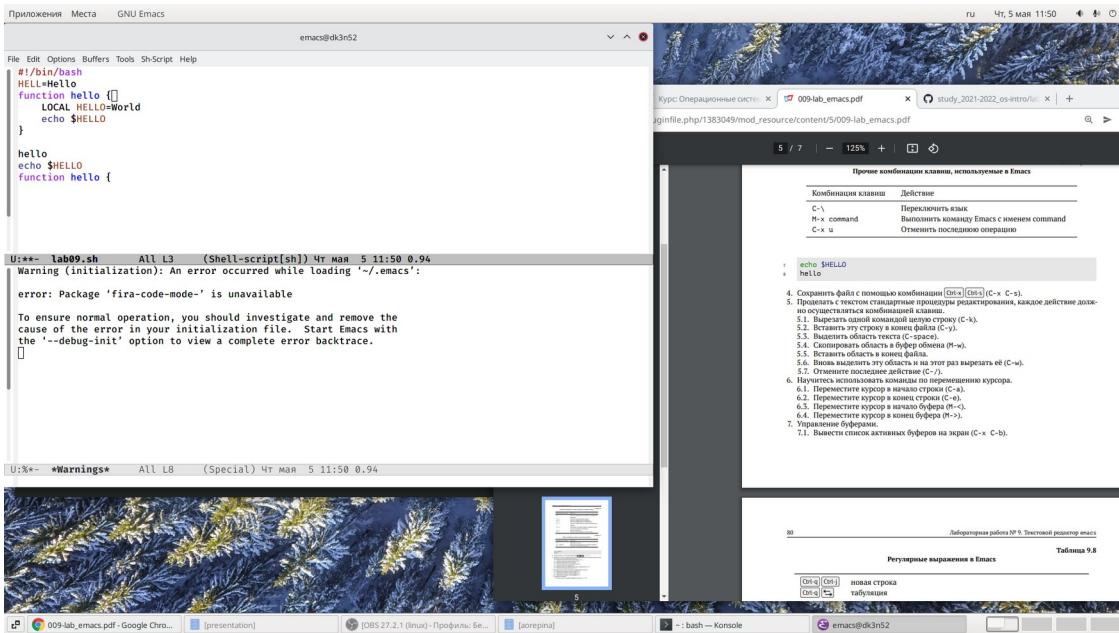
6) Научилась использовать команды по перемещению курсора

а) Переместила курсор в начало строки (Ctrl-a) (рис 9)



9

### б) Переместила курсор в конец строки (Ctrl-e) (рис 10)



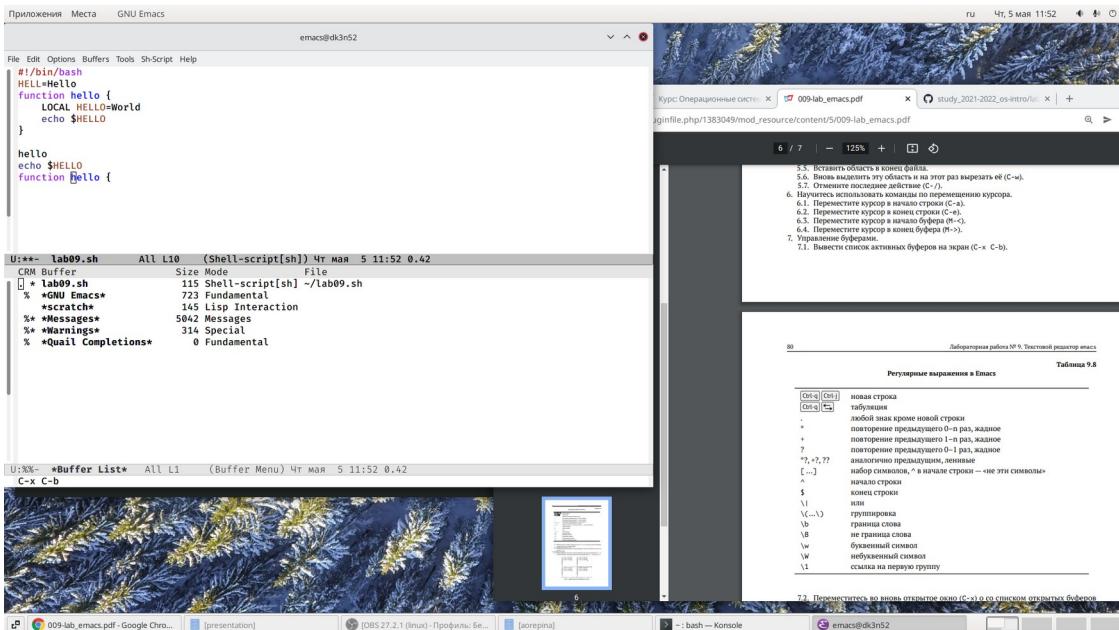
10

### в) Переместила курсор в начало буфера (Alt-<)

### г) Переместила курсор в конец буфера (Alt->)

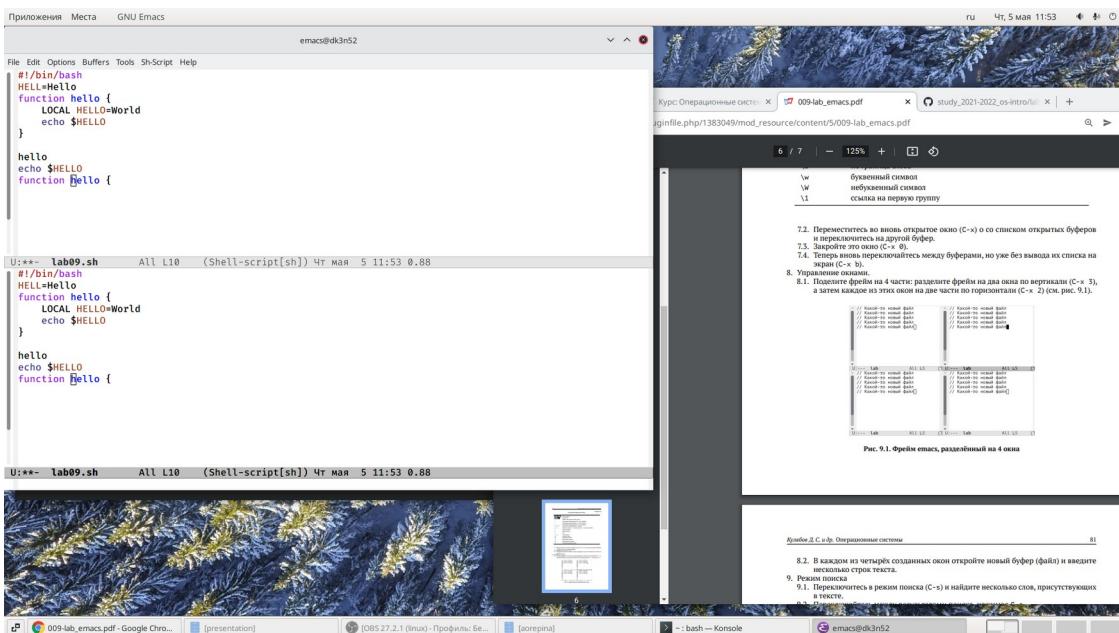
## 7) Научилась управлять буферами

### а) Вывела на экран список активных буферов (Ctrl-x Ctrl-b) (рис 11)



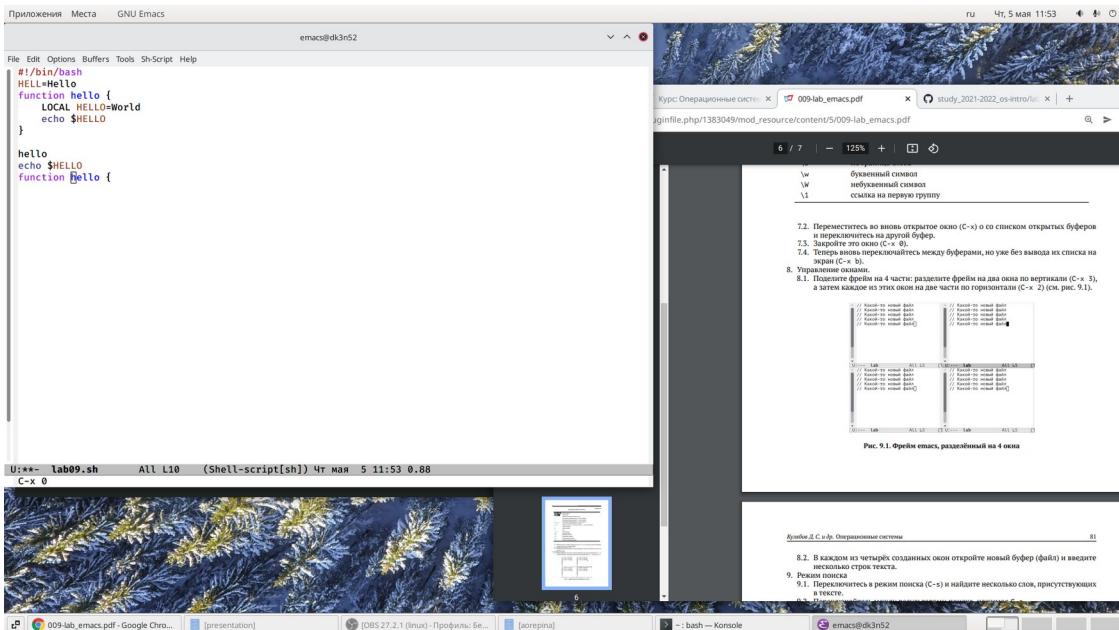
11

б) Переместилась в открытое окно (**Ctrl-x**) со списком открытых буферов и переключилась на другой буфер (рис 12)



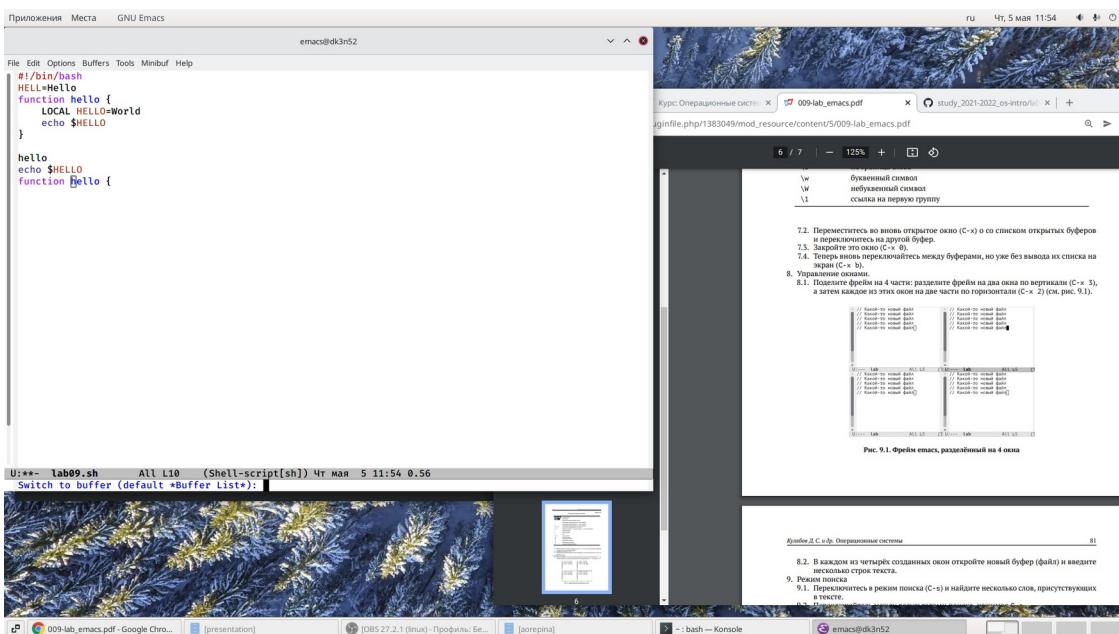
12

в) Закрыла окно (**Ctrl-x 0**) (рис 13)



13

Г) Теперь переключилась между буферами, но уже без вывода их списка на экран (**Ctrl-x b**) (рис 14)



14

8) Научилась управлять окнами

а) Поделила фрейм на 4 части: разделила фрейм на два окна по вертикали (**Ctrl-x 3**) (рис 15) и затем каждое из этих окон по горизонтали на две части (**Ctrl-x 2**) (рис 16)

The screenshot shows a dual-buffer interface in Emacs. Both buffers are titled "lab09.sh" and are of type "Shell-script[sh]". The left buffer contains the following code:

```
#!/bin/bash
HELLO=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

hello
echo $HELLO
function hello {
```

The right buffer contains the same code, with the last line "function hello {" highlighted by a cursor.

15

The screenshot shows four Emacs windows side-by-side. Each window has a title bar indicating it's a Shell-script[sh] buffer. The leftmost window contains the code for lab09.sh, which defines a function hello that prints \$HELLO. The rightmost window contains the code for lab10.sh, which defines a function hello that prints \$LOCAL\_HELLO. The other two windows are partially visible.

```

emacs@dk3n52
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello
function hello {
    LOCAL_HELLO=World
    echo $HELLO
}

hello
echo $HELLO
function hello {[]

U:**- lab09.sh      All L10      (Shell-script[sh])  чт
#!/bin/bash
HELL=Hello
function hello {
    LOCAL_HELLO=World
    echo $HELLO
}

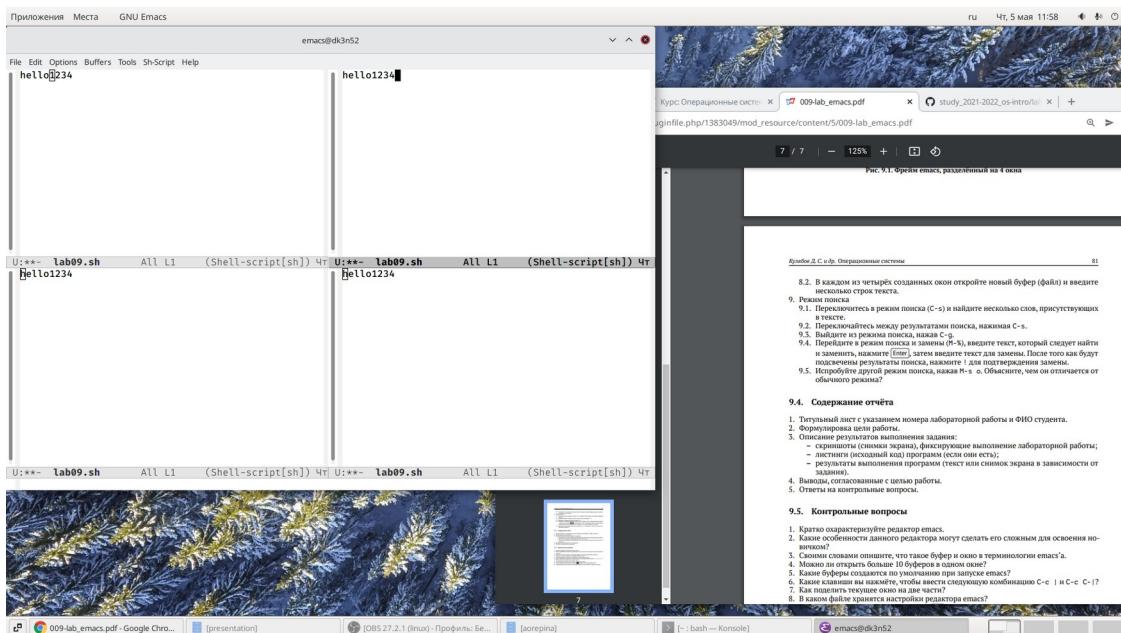
hello
echo $HELLO
function hello {

U:**- lab09.sh      All L10      (Shell-script[sh])  чт
C-x 2
U:**- lab09.sh      All L10      (Shell-script[sh])  чт
U:**- lab09.sh      All L10      (Shell-script[sh])  чт

```

16

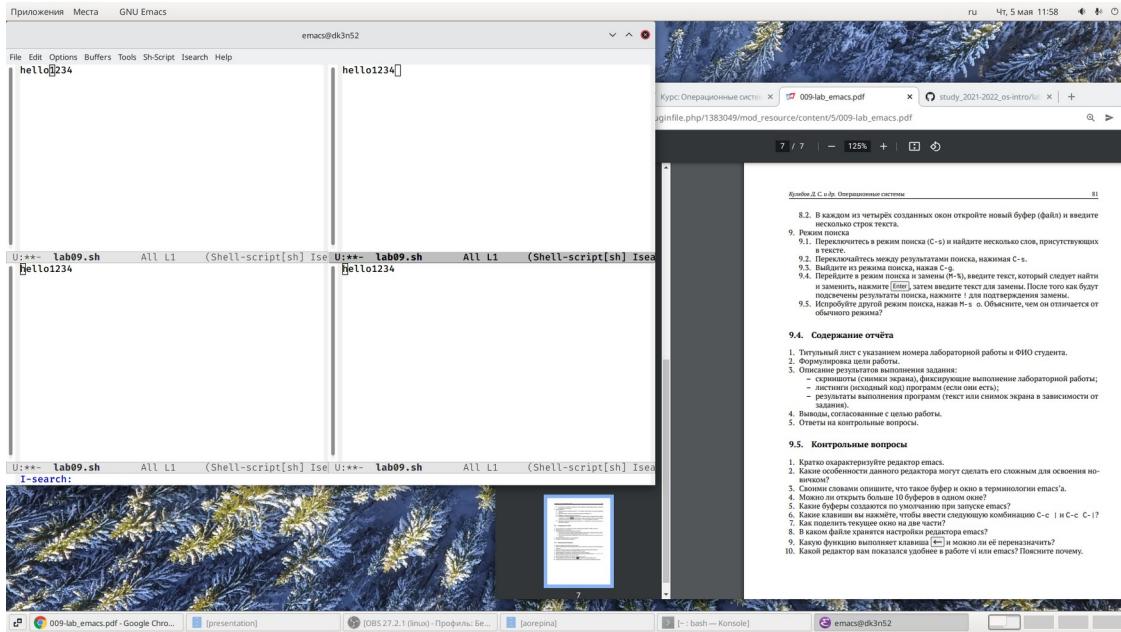
б) В каждом из четырех созданных окон открыла новый буфер и ввела несколько строк текста (рис 17)



17

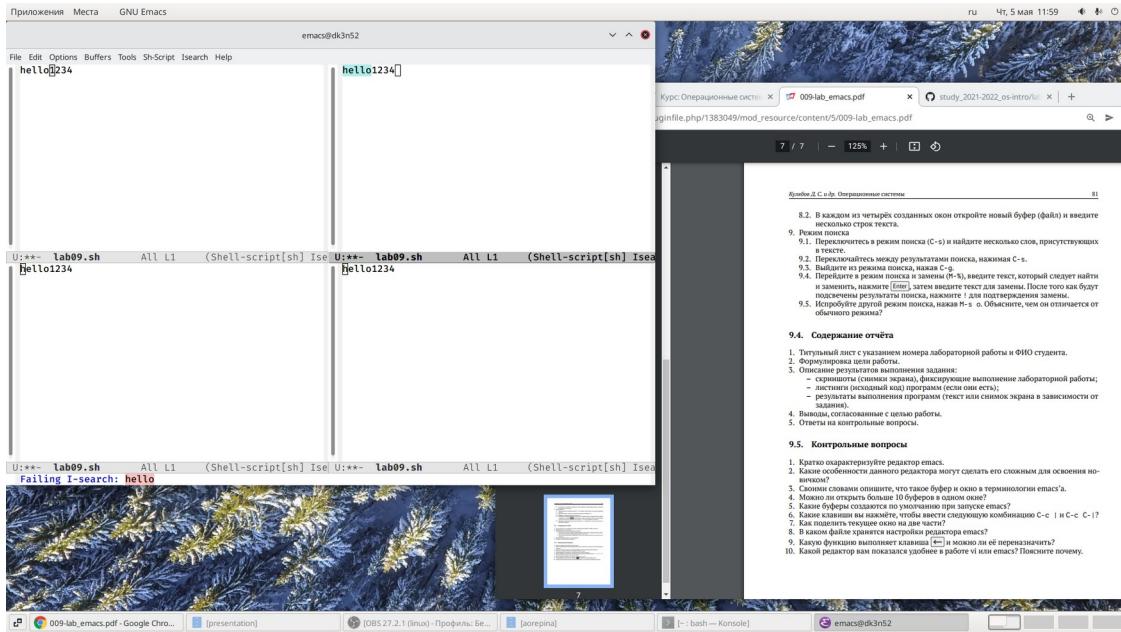
## 9) Режим поиска

а) Переключилась в режим поиска (Ctrl-s) и нашла несколько слов, присутствующих в тексте (рис 18)

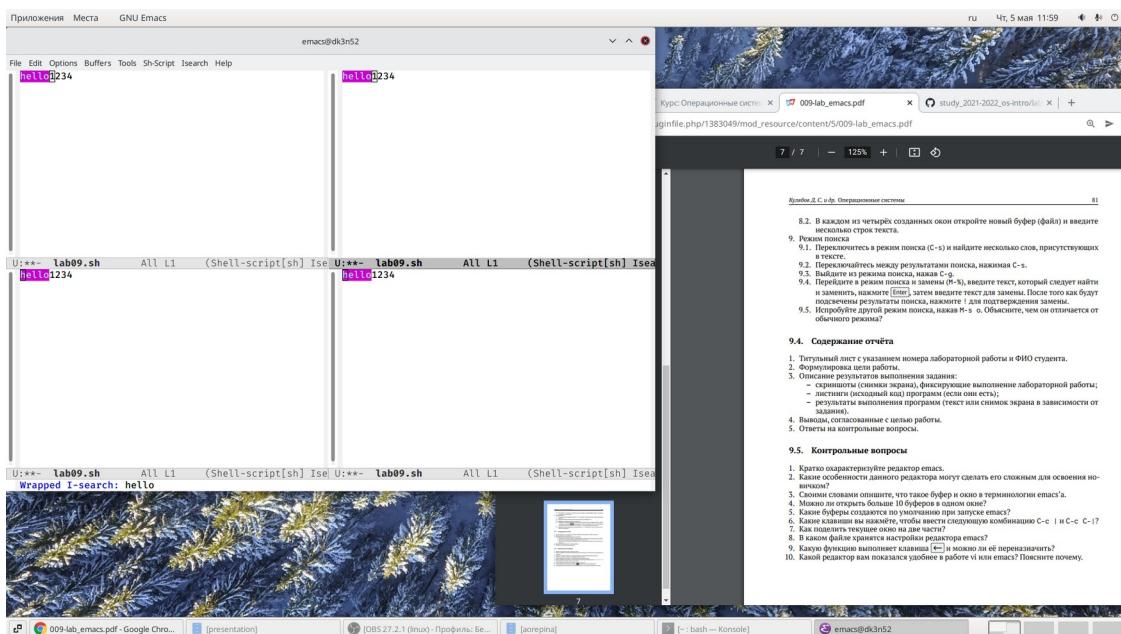


18

б) Переключилась между результатами поиска, нажимая Ctrl-s (рис 19, 20)



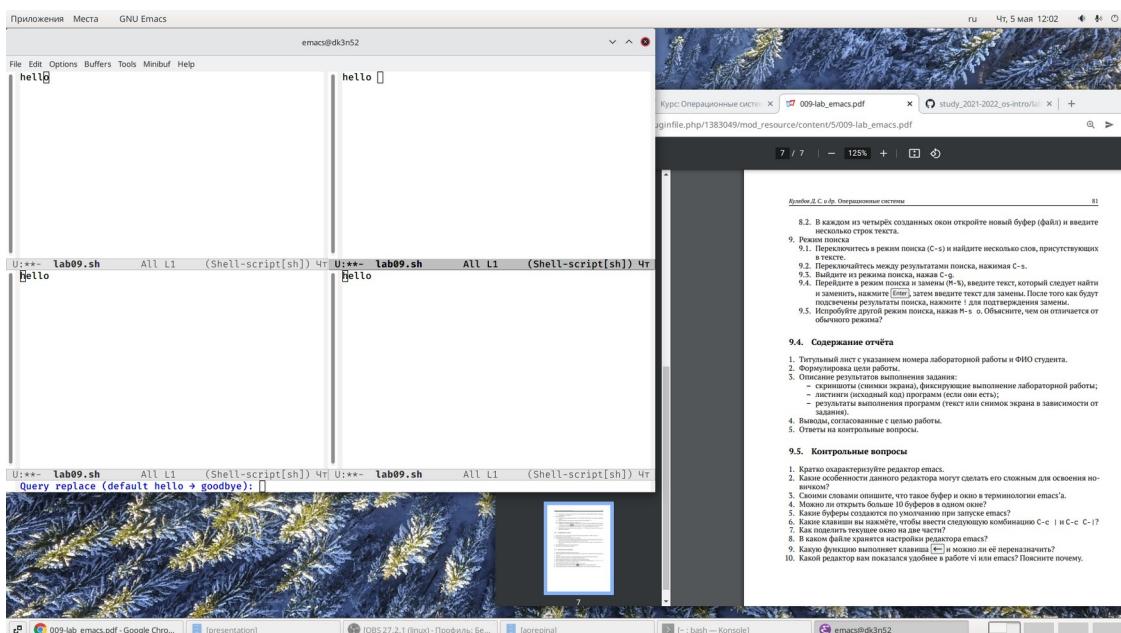
19



20

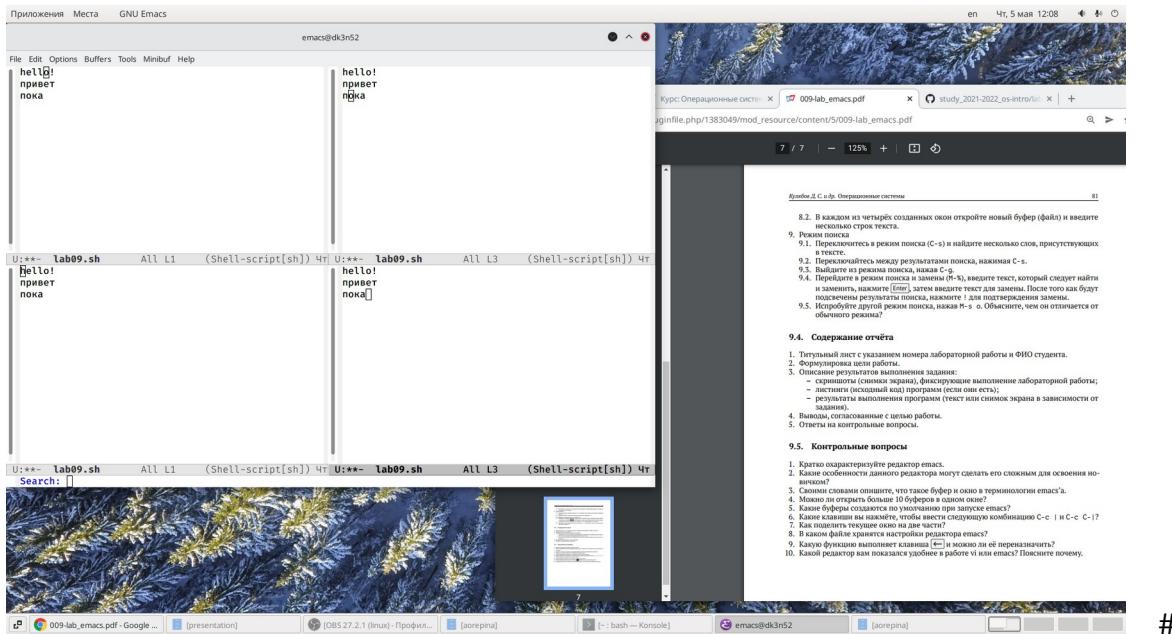
в) Вышла из режима поиска (Ctrl-g)

г) Перешла в режим поиска и замены (Alt-%), ввела текст, который следует найти для замены, нажала Enter, затем ввела текст для замены. После того, как были подсвечены результаты поиска нажала ! для подтверждения замены (рис 21)



21

д) Испробовала другой режим поиска (Alt-s o) (рис 22)



## Выводы

В ходе данной лабораторной работы я познакомилась с операционной системой Linux и получила практические навыки работы с редактором Emacs

## Итоговые вопросы

- 1). Emacs – один из наиболее мощных и широко распространённых редакторов, используемых в мире Unix. По популярности он соперничает с редактором vi и его клонами. В зависимости от ситуации, Emacs может быть текстовым редактором; программой для чтения почты и новостей Usenet; интегрированной средой разработки (IDE); операционной системой и т.д. Всё это разнообразие достигается благодаря архитектуре Emacs, которая позволяет расширять возможности редактора при помощи языка Emacs Lisp. На языке С написаны лишь самые базовые и низкоуровневые части Emacs, включая полнофункциональный интерпретатор языка Lisp. Таким образом, Emacs имеет встроенный язык программирования, который может использоваться для настройки, расширения и изменения поведения редактора. В действительности, большая часть этого редактора, с которым пользователи Emacs работают в наши дни, написана на языке Lisp.
- 2). Основную трудность для новичков при освоении данного редактора могут составлять большое количество команд, комбинаций клавиш, которые не получится все запомнить с первого раза и поэтому придется часто обращаться к справочным материалам.
- 3). Буфер – это объект, представляющий собой текст. Если имеется несколько буферов, то редактировать можно только один. Обычно буфер считывает данные из файла или записывает в файл данные из буфера. Окно – это область экрана, отображающая буфер. При запуске редактора отображается одно окно, но при обращении к некоторым функциям могут открываться дополнительные окна. Окна Emacs окна графической среды XWindow – разные вещи. Одно окно XWindow может быть разбито

на несколько окон в смысле Emacs, в каждом из которых отображается отдельный буфер. 4). Да, можно. 5). При запуске Emacsпо умолчанию создаются следующие буфера: «scratch»(буфер для несохраненного текста) «Messages»(журнал ошибок, включающий также информацию, которая появляется в области EchoArea) «GNUEmacs»(справочный буфер о редакторе). 6). С-с |сначала, удерживая «ctrl», нажимаю «с», после –отпускаю обе клавиши нажимаю «|» С-сС-|сначала, удерживая «ctrl», нажимаю «с», после –отпускаю обе клавиши и, удерживая «ctrl», нажимаю «|». 7). Чтобы поделить окно на две части необходимо воспользоваться комбинацией «Ctrl-x 3»(по вертикали) или «Ctrl-x 2» (по горизонтали). 8). Настройки Emacsхранятся в файле .emacs. 9). По умолчанию клавиша «←» удаляет символ перед курсором, нов редакторе её можно переназначить. Для этого необходимо изменить конфигурацию файла .emacs. 10). Более удобным я считаю редактор emacs, потому что в нем проще открывать другие файлы, можно использовать сразу несколько окон, нет «Командногорежима», «Режима ввода», «Режима командной строки», которые являются немного непривычными и в какой-то степени неудобным.