

# **Лабораторная работа №10**

**Дисциплина: Операционные системы**

**Егорова Александра**

# **Содержание**

<b>1 Цель работы</b>	<b>3</b>
<b>2 Задание</b>	<b>4</b>
<b>3 Выполнение лабораторной работы</b>	<b>5</b>
<b>4 Выводы</b>	<b>26</b>
<b>5 Контрольные вопросы</b>	<b>27</b>

# **1 Цель работы**

Познакомиться с операционной системой Linux. Получить практические навыки работы с редактором Emacs.

## **2 Задание**

1. Ознакомиться с теоретическим материалом.
2. Ознакомиться с редактором emacs.
3. Выполнить упражнения.
4. Ответить на контрольные вопросы

# 3 Выполнение лабораторной работы

Для работы с Emacs необходимо установить данный редактор, используя команду “sudo yum install emacs”. (рис. -fig. 3.1)

```
aegorova@aegorova:~$ sudo yum install emacs
[sudo] пароль для аегорова:
Загружены модули: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror.hoster.kz
 * extras: mirror.hoster.kz
 * updates: mirror.hoster.kz
Разрешение зависимостей
--> Проверка сценария
--> Пакет emacs.x86_64 1:24.3-23.el7 помечен для установки
--> Обработка зависимостей: emacs-common = 1:24.3-23.el7 пакета: 1:emacs-24.3-23.el7.x86_64
--> Обработка зависимостей: libotf.so.0()(64bit) пакета: 1:emacs-24.3-23.el7.x86_64
--> Обработка зависимостей: libMagickWand-6.016.so.6()(64bit) пакета: 1:emacs-24.3-23.el7.x86_64
--> Обработка зависимостей: libMagickCore-6.016.so.6()(64bit) пакета: 1:emacs-24.3-23.el7.x86_64
--> Обработка зависимостей
--> Пакет ImageMagick.x86_64 0:6.9.10.68-5.el7_9 помечен для установки
--> Обработка зависимостей: libwmflite-0.2.so.7()(64bit) пакета: ImageMagick-6.9.10.68-5.el7_9.x86_64
--> Пакет emacs-common.x86_64 1:24.3-23.el7 помечен для установки
--> Обработка зависимостей: liblockfile.so.1()(64bit) пакета: 1:emacs-common-24.3-23.el7.x86_64
--> Проверка сценария
--> Пакет libwmf-lite.x86_64 0:0.2.8.4-44.el7 помечен для установки
--> Проверка зависимостей окончена

Зависимости определены
=====
|-----[Package]-----|-----[Архитектура]-----|-----[Версия]-----|-----[Репозиторий]-----|-----[Размер]-----| |
| Установка: | emacs | x86_64 | 1:24.3-23.el7 | base | 2.9 M |
| Установка зависимостей: | ImageMagick | x86_64 | 6.9.10.68-5.el7_9 | updates | 2.3 M |
| emacs-common | x86_64 | 1:24.3-23.el7 | base | 20 M |
| liblockfile | x86_64 | 1.88-17.el7 | base | 21 k |
| libotf | x86_64 | 0:0.9.13-4.el7 | base | 98 K |
| libwmf-lite | x86_64 | 0:0.2.8.4-44.el7 | base | 66 K |
```

Figure 3.1: Установка emacs

- 1) Откроем редактор Emacs с помощью команды «emacs &». (рис. -fig. 3.2) (рис. -fig. 3.3)

```
aegorova@aegorova:~$ Файл Правка Вид Поиск Терминал Справка
[aegorova@aegorova ~]$ emacs &
[1] 9602
```

Figure 3.2: Редактор Emacs

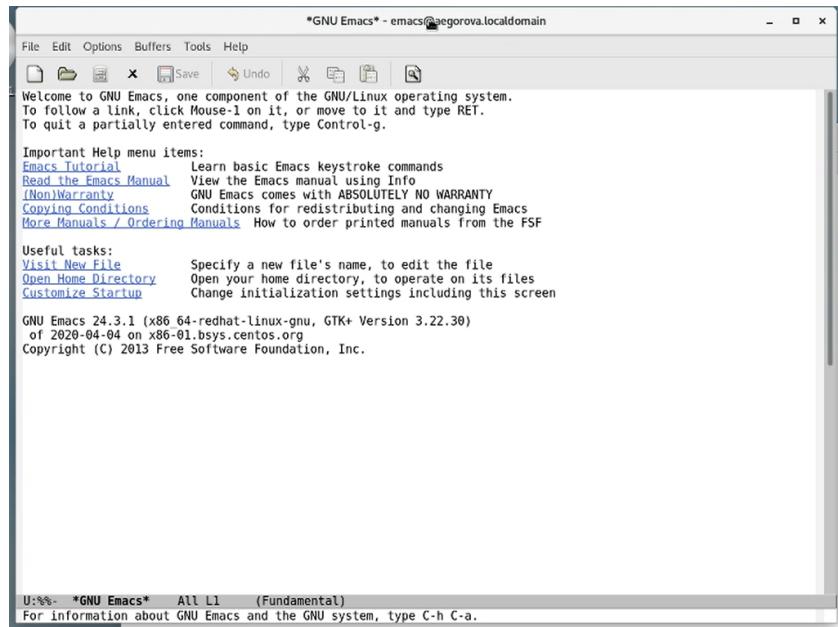


Figure 3.3: Редактор Emacs

- 2) Создадим файл lab10.sh с помощью комбинации «Ctrl-x» «Ctrl-f». (рис. - fig. 3.4)

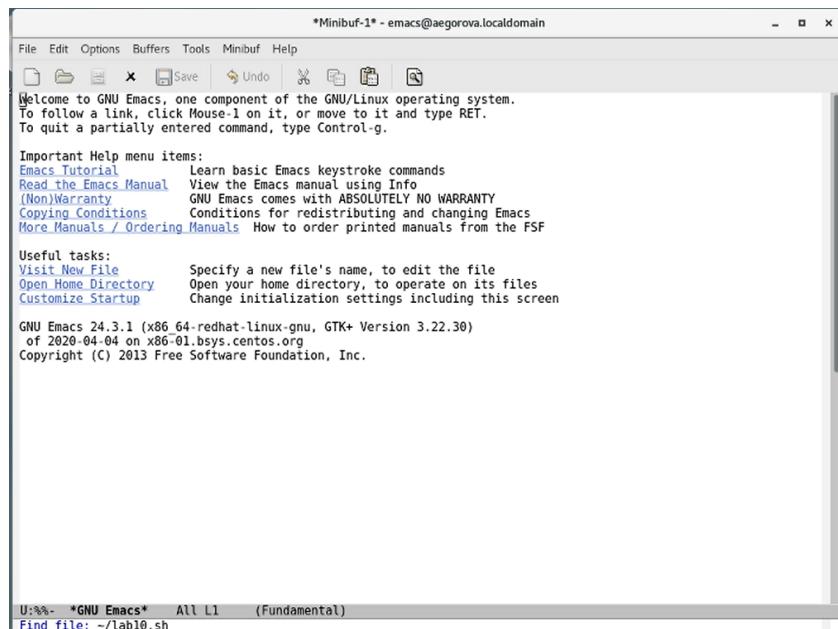


Figure 3.4: Создаем файл

- 3) В открывшемся буфере наберем необходимый текст. (рис. -fig. 3.5)

The screenshot shows a terminal window titled "lab". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". The toolbar contains icons for file operations like Open, Save, Undo, and Cut/Paste. The main text area displays the following shell script:

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

Figure 3.5: Вводим текст

- 4) Сохраним файл с помощью комбинации «Ctrl-x» «Ctrl-s». 5) Проделаем с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш. 5.1) Вырежем одной командой целую строку («Ctrl-k»). (рис. -fig. 3.6)

The screenshot shows a terminal window titled "lab10.sh". The menu bar and toolbar are identical to Figure 3.5. The main text area shows the same shell script as Figure 3.5, but the line "LOCAL HELLO=World" is highlighted with a selection bar underneath it.

Figure 3.6: Вырезаем строку

- 5.2) Вставим эту строку в конец файла («Ctrl-y»). (рис. -fig. 3.7)

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

echo $HELLO
```

Figure 3.7: Вставляем строку

5.3) Выделим область текста («Ctrl-space»). (рис. -fig. 3.8)

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

echo $HELLO
```

Figure 3.8: Выделим область

5.4) Скопируем область в буфер обмена («Alt-w»). 5.5) Вставим область в конец файла («Ctrl-y»). (рис. -fig. 3.9)

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

echo $HELLO
echo $HELLO
```

Figure 3.9: Вставим область

5.6) Вновь выделим эту область («Ctrl-space») (Рисунок 9) и на этот раз вырежем её («Ctrl-w»). (рис. -fig. 3.10)

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

echo $HELLO
echo $HELLO
```

Figure 3.10: Скопирем и вставим область

5.7) Отменим последнее действие («Ctrl-x u»). (рис. -fig. 3.11)

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

echo $HELL
echo $LOCAL
```

Figure 3.11: Отмена последнего действия

- 6) Научимся использовать команды по перемещению курсора. 6.1) Переместим курсор в начало строки («Ctrl-a»). (рис. -fig. 3.12)

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

echo $HELLO
echo $HELLO
```

Figure 3.12: Перемещение курсора

6.2) Переместим курсор в конец строки («Ctrl-e»). (рис. -fig. 3.13)

The screenshot shows a window titled "lab10.sh - emacs". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". Below the menu is a toolbar with icons for file operations and a "Save" button. The main buffer contains a shell script:

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

echo $HELLO
echo $HELLO
```

A cursor is positioned at the end of the second "echo" command. The status bar at the bottom right shows "line 12 column 19".

Figure 3.13: Перемещение курсора

6.3) Переместим курсор в начало буфера («Alt-<»). (рис. -fig. 3.14)

The screenshot shows a terminal window titled "lab10.sh - e". The window has a menu bar with "File", "Edit", "Options", "Buffers", "Tools", and "Sh-Scrip". Below the menu is a toolbar with icons for file operations: a blank document, a folder, a file, a close button, a save icon labeled "Save", and an undo icon labeled "Und". The main area of the window contains the following shell script code:

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

echo $HELLO
echo $HELLO
```

Figure 3.14: Перемещение курсора

6.4) Переместим курсор в конец буфера («Alt->»). (рис. -fig. 3.15)

The screenshot shows a window titled "lab10.sh - emacs". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". Below the menu is a toolbar with icons for file operations (New, Open, Save, Undo) and a "Save" button. The main buffer contains the following Bash script:

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

echo $HELLO
echo $HELLO
```

Figure 3.15: Перемещение курсора

- 7) Управление буферами. 7.1) Выведем список активных буферов на экран («Ctrl-x» «Ctrl-b»). (рис. -fig. 3.16)

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

echo $HELLO
echo $HELLO
```

U:-- lab10.sh	All L10	(Shell-script[sh])
CRM Buffer	Size	Mode
0 * lab10.sh	102	Shell-script[sh] ~/lab10.sh
* Messages*	1997	Fundamental
% *GNU Emacs*	901	Fundamental
*scratch*	191	Lisp Interaction

U:%%- \*Buffer List\* All L1 (Buffer Menu)  
C-x C-b

Figure 3.16: Список активных буферов

7.2) Переместимся во вновь открытое окно («Ctrl-x o») со списком открытых буферов и переключимся на другой буфер (для этого необходимо нажать на «enter» после выбора необходимого буфера). (рис. -fig. 3.17)

The screenshot shows the Emacs Buffer List window titled "\*Buffer List\* - emacs@aegorova.localdomain". The window contains a menu bar with File, Edit, Options, Buffers, Tools, Buffer-Menu, and Help. Below the menu is a toolbar with icons for Save, Undo, and others. The main area displays a shell script named "lab10.sh" with the following content:

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

echo $HELLO
echo ${HELLO[]}
```

Below the script, a buffer list table is shown:

CRM Buffer	Size	Mode	File
.* lab10.sh	102	Shell-script[sh]	~/lab10.sh
*Messages*	1997	Fundamental	
% *GNU Emacs*	901	Fundamental	
*scratch*	191	Lisp Interaction	

The status bar at the bottom shows "U:%%- \*Buffer List\* All L1 (Buffer Menu)" and "C-x o".

Figure 3.17: Список активных буферов

7.3) Закроем это окно («Ctrl-x 0»). (рис. -fig. 3.18)

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

echo $HELLO
echo $HELLO
```

Figure 3.18: Закрываем окно

7.4) Теперь вновь переключимся между буферами, но уже без вывода их списка на экран («Ctrl-x b»). (рис. -fig. 3.19) (рис. -fig. 3.20)



Figure 3.19: Переключимся между буферами

```
*scratch* - emacs@aegorova.loca...
File Edit Options Buffers Tools Lisp-Interaction Help
;; This buffer is for notes you don't want to save, and for Lisp evaluation.
```

Figure 3.20: Переключимся между буферами

8) Управление окнами 8.1) Поделим фрейм на 4 части: разделим фрейм на

два окна по вертикали («Ctrl-x 3»), а затем каждое из этих окон на две части по горизонтали («Ctrl-x 2»). (рис. -fig. 3.21) (рис. -fig. 3.22) (рис. -fig. 3.23)

The screenshot shows the Emacs interface with two vertically split frames. The top frame contains the code for a shell script named 'lab10.sh'. The bottom frame is a 'Messages' buffer showing the output of running the script: four 'byte-code: Beginning of buffer' messages followed by the command 'C-x 3'.

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

echo $HELLO
echo $HELLO[]
```

U:++- lab10.sh All L10 (Shell-script[sh]) U:++- lab10.sh All L10 (Shell-script[sh])
byte-code: Beginning of buffer [4 times]
byte-code: Beginning of buffer[]
U:++- \*Messages\* Bot L63 (Fundamental)
C-x 3

Figure 3.21: Разделение фрейма

The screenshot shows the Emacs interface with two horizontally split frames. The left frame contains the code for a shell script named 'lab10.sh'. The right frame is a 'Messages' buffer showing the output of running the script: four 'byte-code: Beginning of buffer' messages followed by the command 'C-x 2'.

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

echo $HELLO
echo $HELLO[]
```

U:++- lab10.sh All L10 (Shell-script[sh])
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo \$HELLO
}
hello

echo \$HELLO
echo \$HELLO[]

U:++- lab10.sh All L10 (Shell-script[sh]) U:++- lab10.sh All L10 (Shell-script[sh])
byte-code: Beginning of buffer [4 times]
byte-code: Beginning of buffer[]
U:++- \*Messages\* Bot L63 (Fundamental)
C-x 2

Figure 3.22: Разделение фрейма

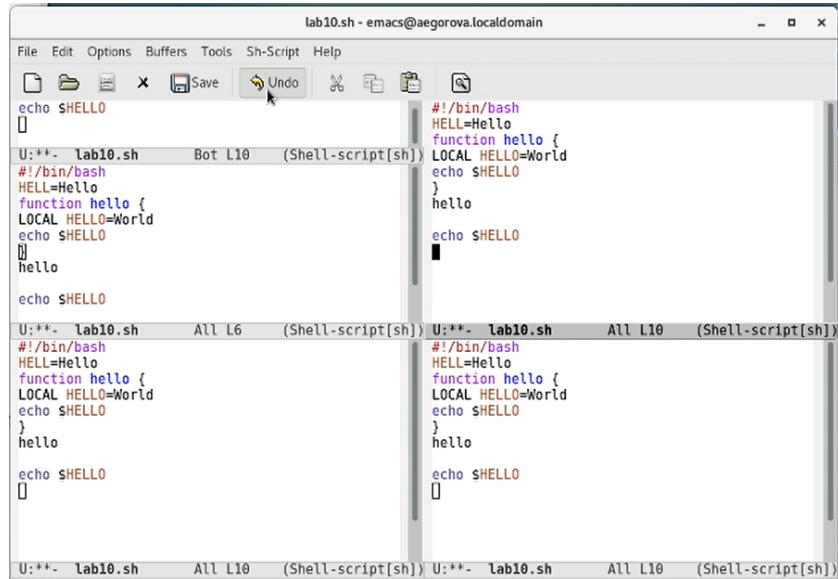


Figure 3.23: Разделение фрейма

8.2) В каждом из четырёх созданных окон откроем новый буфер (файл) и введем несколько строк текста. Для этого предварительно создадим эти файлы с помощью команд «`touch example1.txt`», «`touch example2.txt`», «`touch example3.txt`», «`touch example4.txt`». (рис. -fig. 3.24) (рис. -fig. 3.25)

```
[aegorova@aegorova ~]$ touch example1.txt
[aegorova@aegorova ~]$ touch example2.txt
[aegorova@aegorova ~]$ touch example3.txt
[aegorova@aegorova ~]$ touch example4.txt
[aegorova@aegorova ~]$
```

Figure 3.24: Создаем файлы

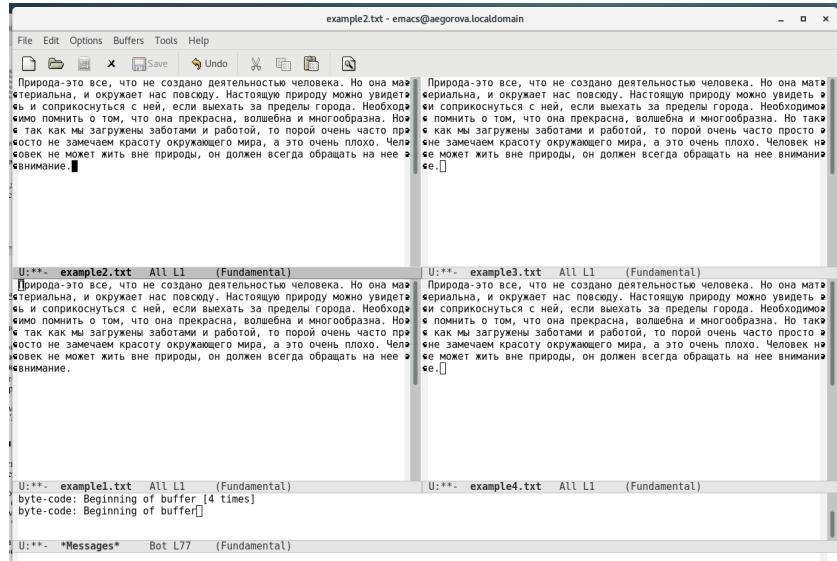


Figure 3.25: Вводим текст

9) Режим поиска 9.1) Переключимся в режим поиска («Ctrl-s») и найдем несколько слов, присутствующих в тексте. (рис. -fig. 3.26) (рис. -fig. 3.27)

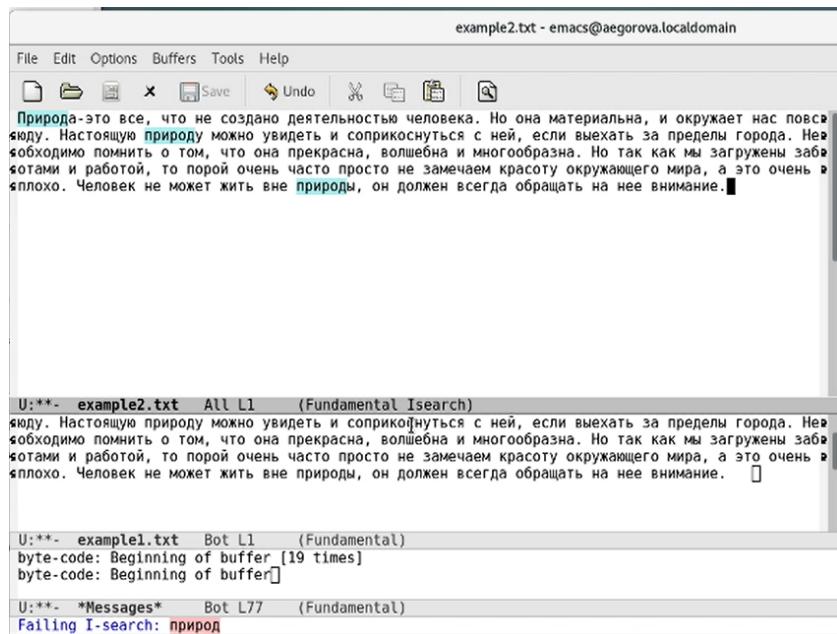


Figure 3.26: Переключимся в режим поиска

The screenshot shows the Emacs interface with the buffer `example2.txt` open. The main window displays the text: "Природа-это все, что не создано деятельностью человека. Но она материальна, и окружает нас повсюду. Настоящую природу можно увидеть и соприкоснуться с ней, если выехать за пределы города. Несободимо помнить о том, что она прекрасна, волшебна и многообразна. Но так как мы загружены заботами и работой, то порой очень часто просто не замечаем красоту окружающего мира, а это очень плохо. Человек не может жить вне природы, он должен всегда обращать на нее внимание." A search highlight for the word "природ" is visible in the text.

Below the main window, three search results are listed in the minibuffer:

- U:\*\*- example2.txt All L1 (Fundamental Isearch)
- byte-code: Beginning of buffer [19 times]
- byte-code: Beginning of buffer[]
- U:\*\*- example1.txt Bot L1 (Fundamental)
- U:\*\*- \*Messages\* Bot L77 (Fundamental)
- Wrapped I-search: природ

Figure 3.27: Найдем несколько слов

9.2) Переключимся между результатами поиска, нажимая «Ctrl-s». (рис. -fig. 3.28)

This screenshot is identical to Figure 3.27, showing the Emacs interface with the buffer `example2.txt` open and the minibuffer displaying search results for the word "природ".

Figure 3.28: Переключимся между результатами поиска

9.3) Выйдем из режима поиска, нажав «Ctrl-g». (рис. -fig. 3.29)

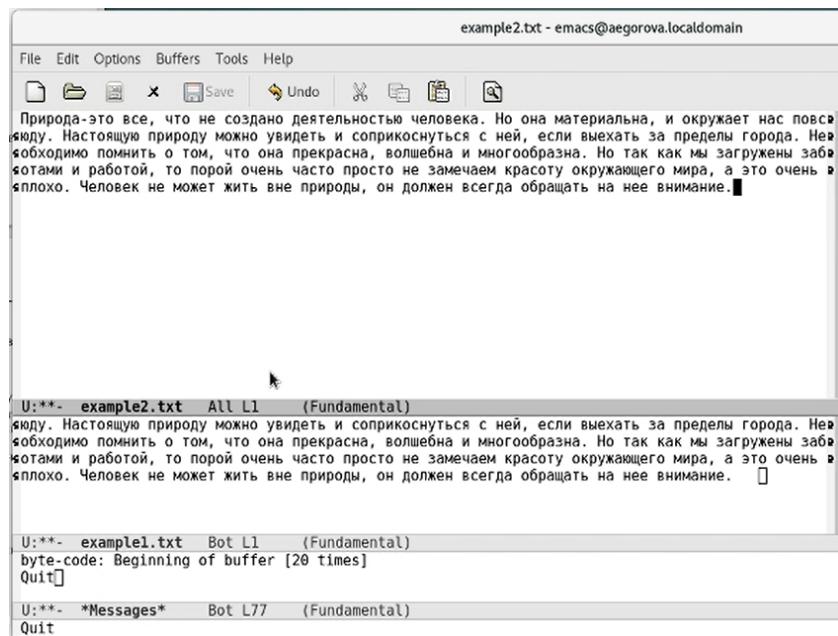


Figure 3.29: Выйдем из режима поиска

9.4) Перейдем в режим поиска и замены («Alt-%»), введем текст, который следует найти и заменить, нажмем «enter», затем введем текст для замены. После того как будут подсвечены результаты поиска, нажмем «!» для подтверждения замены. Важно, чтобы курсор находился в начале текста. (рис. -fig. 3.30) (рис. -fig. 3.31) (рис. -fig. 3.32) (рис. -fig. 3.33)

The screenshot shows the Emacs interface with multiple buffers open. The main buffer (Minibuf-1) contains a paragraph about nature. A search and replace operation is in progress, with the word 'природа' highlighted in red. The message bar at the bottom indicates the command: 'Query replace (default природа -> мир): природа'. Other buffers visible include 'example2.txt' and 'example1.txt'.

Figure 3.30: Замена текста

The screenshot shows the Emacs interface after the replacement has been completed. The word 'природа' has been replaced by 'мир' throughout the main buffer. The message bar at the bottom now shows the command: 'Query replace природа with: мир'.

Figure 3.31: Замена текста

example2.txt - emacs@aegorova.localdomain

File Edit Options Buffers Tools Help

Save Undo

Природа-это все, что не создано деятельностью человека. Но она материальна, и окружает нас повсюду. Настоящую природу можно увидеть и соприкоснуться с ней, если выехать за пределы города. Необходимо помнить о том, что она прекрасна, волшебна и многообразна. Но так как мы загружены заботами и работой, то порой очень часто просто не замечаем красоту окружающего мира, а это очень плохо. Человек не может жить вне природы, он должен всегда обращать на нее внимание.

U:\*\*- example2.txt All L1 (Fundamental)

Mark set [2 times]

U:\*\*- \*Messages\* Bot L100 (Fundamental)

Query replacing природа with Мир: (? for help)

Figure 3.32: Замена текста

example2.txt - emacs@aegorova.localdomain

File Edit Options Buffers Tools Help

Save Undo

Мир-это все, что не создано деятельностью человека. Но она материальна, и окружает нас повсюду. Настоящую природу можно увидеть и соприкоснуться с ней, если выехать за пределы города. Необходимо помнить о том, что она прекрасна, волшебна и многообразна. Но так как мы загружены заботами и работой, то порой очень часто просто не замечаем красоту окружающего мира, а это очень плохо. Человек не может жить вне природы, он должен всегда обращать на нее внимание.

U:\*\*- example2.txt All L1 (Fundamental)

Mark set [2 times]

U:\*\*- \*Messages\* Bot L100 (Fundamental)

Replaced 1 occurrence

Figure 3.33: Замена текста

9.5) Пробуем другой режим поиска, нажав «Alt-s o». (рис. -fig. 3.34) (рис. -fig. 3.35)

\*Minibuf-1\* - emacs@aegorova.localdomain

File Edit Options Buffers Tools Minibuf Help

Save Undo

Мир[это все, что не создано деятельности человека. Но она материальна, и окружает нас повсюду.]  
с Настоящую природу можно увидеть и соприкоснуться с ней, если выехать за пределы города. Необходимо помнить о том, что она прекрасна, волшебна и многообразна. Но так как мы загружены заботами и работой, то порой очень часто просто не замечаем красоту окружающего мира, а это очень плохо. Человек не может жить вне природы, он должен всегда обращать на нее внимание.

U:\*\*- example2.txt All L1 (Fundamental)

году. Настоящую природу можно увидеть и соприкоснуться с ней, если выехать за пределы города. Необходимо помнить о том, что она прекрасна, волшебна и многообразна. Но так как мы загружены заботами и работой, то порой очень часто просто не замечаем красоту окружающего мира, а это очень плохо. Человек не может жить вне природы, он должен всегда обращать на нее внимание. □

U:\*\*- example1.txt Bot L1 (Fundamental)

Replaced 1 occurrence

[]

U:\*\*- \*Messages\* Bot L101 (Fundamental)

List lines matching regexp:

Figure 3.34: Другой режим поиска

example2.txt - emacs@aegorova.localdomain

File Edit Options Buffers Tools Help

Save Undo

Мир[это все, что не создано деятельности человека. Но она материальна, и окружает нас повсюду.]  
с Настоящую природу можно увидеть и соприкоснуться с ней, если выехать за пределы города. Необходимо помнить о том, что она прекрасна, волшебна и многообразна. Но так как мы загружены заботами и работой, то порой очень часто просто не замечаем красоту окружающего мира, а это очень плохо. Человек не может жить вне природы, он должен всегда обращать на нее внимание.

U:\*\*- example2.txt All L1 (Fundamental)

сплохо. Человек не может жить вне природы, он должен всегда обращать на нее внимание. □

U:\*\*- example1.txt Bot L1 (Fundamental)

1 match for "и" in buffer: example2.txt

□ 1:Мир[это все, что не создано деятельности человека. Но она материальна, и окружает нас повсюду.]  
и соприкоснуться с ней, если выехать за пределы города. Необходимо помнить о том, что она прекрасна, как мы загружены заботами и работой, то порой очень часто просто не замечаем красоту окружающего мира, а это очень плохо. Человек не может жить вне природы, он должен всегда обращать на нее внимание.

U:%- \*Occur\* All L2 (Occur)

Figure 3.35: Другой режим поиска

## **4 Выводы**

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

## 5 Контрольные 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: а) текстовым редактором; б) программой для чтения почты и новостей Usenet; в) интегрированной средой разработки (IDE); г) операционной системой и т.д. Всё это разнообразие достигается благодаря архитектуре Emacs, которая позволяет расширять возможности редактора при помощи языка Emacs Lisp. На языке C 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 того 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 и окна графической среды X Window – разные вещи. Одно окно X Window может быть разбито на несколько окон в смысле Emacs, в каждом из которых отображается отдельный буфер.

- 4) Да, можно.
- 5) При запуске Emacs по умолчанию создаются следующие буферы: а) «scratch» (буфер для несохраненного текста); б) «Messages» (журнал ошибок, включающий также информацию, которая появляется в области EchoArea); в) «GNU Emacs» (справочный буфер о редакторе).
- 6) С-с | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и нажимаю «|». С-с С-| 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 Emacs хранятся в файле .emacs.
- 9) По умолчанию клавиша «» удаляет символ перед курсором, но в 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 «Командного режима», «Режима ввода», «Режима командной строки», которые являются немного непривычными и в какой-то степени неудобными.