

Лабораторная работа №10

Дисциплина: Операционные системы

Коновалова Татьяна Борисовна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Библиография	40
5	Выводы	41

List of Tables

List of Figures

3.1	Открытие редактора	7
3.2	Создание файла	8
3.3	Работа с текстом	8
3.4	Вырезание строки	9
3.5	Вставила строку	10
3.6	Выделила область текста	11
3.7	Область в конце файла	12
3.8	Выделила область	13
3.9	Вырезание строки	14
3.10	Отмена последнего действия	15
3.11	Переместила курсор	16
3.12	Перемещение курсора	17
3.13	Перемещение курсора	18
3.14	Перемещение курсора	19
3.15	Перемещение курсора	20
3.16	Список активных буферов	21
3.17	Список открытых буферов	22
3.18	Переключение на буфера	23
3.19	Закрыла окно буфера	24
3.20	Переключение между буферами	25
3.21	Работа с фреймом	26
3.22	Работа с фреймом	27
3.23	Работа с фреймом	28
3.24	Создание файлов	28
3.25	Команда для работы с файлами	29
3.26	Вводим текст в документы	30
3.27	Режим поиска	31
3.28	Поиск слов в тексте	32
3.29	Переключение между поисками	33
3.30	Переключение между поисками	34
3.31	Выходим из режима поиска	35
3.32	Переходим в режим поиска	36
3.33	Переходим в режим поиска	37
3.34	Другой режим поиска	38

1 Цель работы

Цель данной лабораторной работы — Познакомиться с операционной системой Linux. Получить практические навыки работы с редактором Emacs.

2 Задание

1. Сделать отчёт по лабораторной работе №10 в формате Markdown.
2. Познакомиться с операционной системой Linux.

3 Выполнение лабораторной работы

1). Открыла редактор Emacs с помощью команды «emacs &» (Скриншот 3.1).

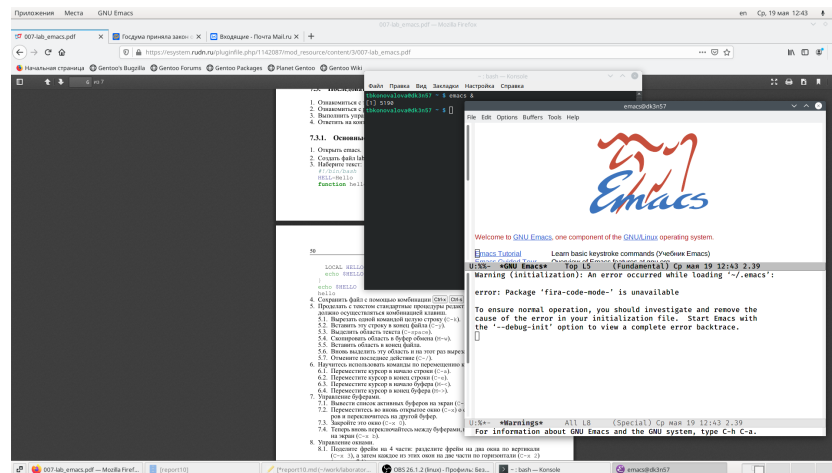


Figure 3.1: Открытие редактора

2). Создала файл lab07.sh с помощью комбинации «Ctrl-x» «Ctrl-f» (Скриншот 3.2).

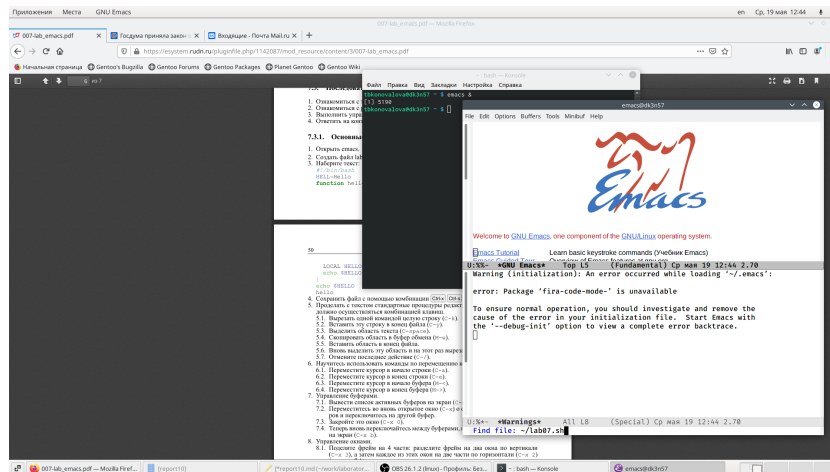


Figure 3.2: Создание файла

3). В открывшемся буфере набрала необходимый текст (алгоритм действий представлен на рис. 3.3).

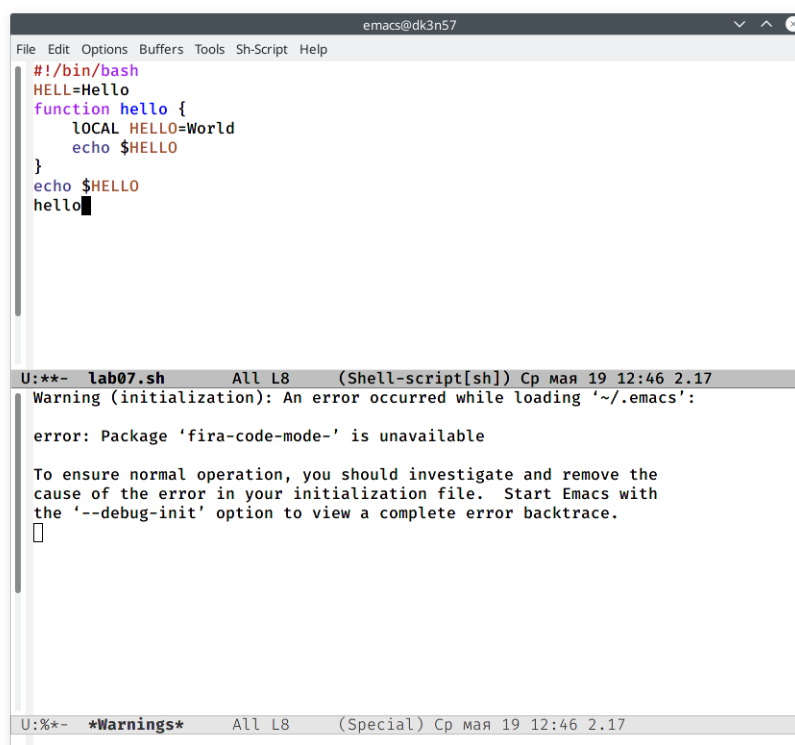
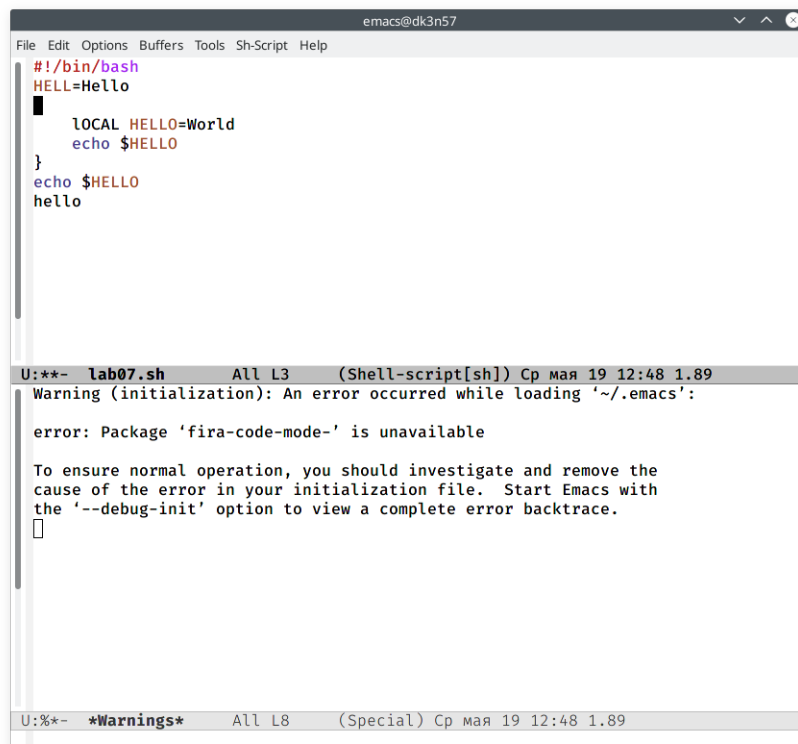


Figure 3.3: Работа с текстом

4). Сохранила файл с помощью комбинации «Ctrl-x»«Ctrl-s».

5). Выполнила следующие действия:

5.1. Вырезала одной командой целую строку («Ctrl-k») (См. рисунок 3.4).



The screenshot shows an Emacs editor window titled 'emacs@dk3n57'. The main editing area contains a shell script with the following content:

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

Below the script, a status bar indicates 'U:***- lab07.sh All L3 (Shell-script[sh]) Ср мая 19 12:48 1.89'. Below this, a warning message is displayed:

```
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, another status bar shows 'U:%*- *Warnings* All L8 (Special) Ср мая 19 12:48 1.89'.

Figure 3.4: Вырезание строки

5.2. Вставила эту строку в конец файла («Ctrl-y») (См. рисунок 3.5).

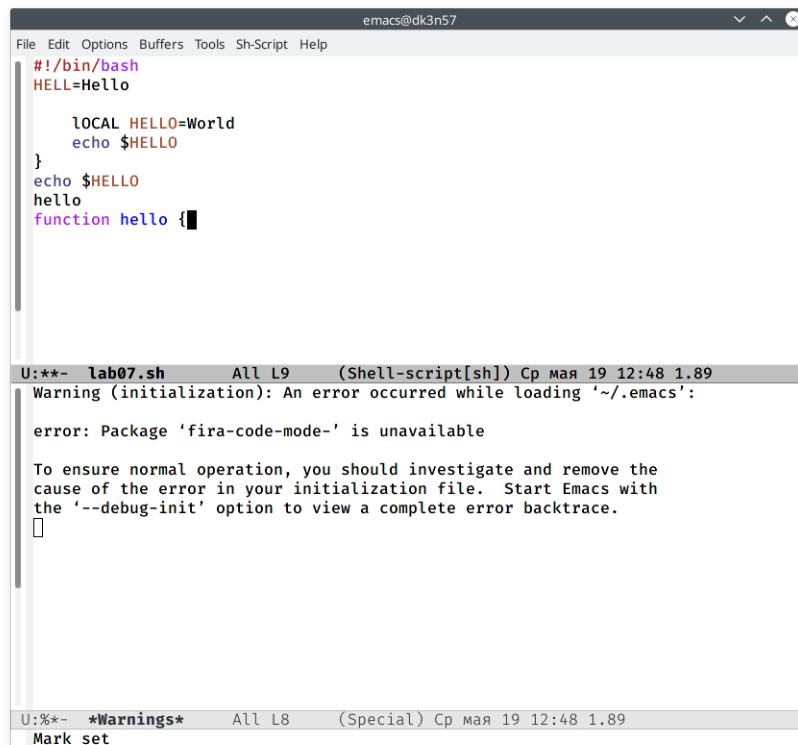


Figure 3.5: Вставила строку

5.3. Выделила область текста («Ctrl-space») (алгоритм действий представлен на рис. 3.6).

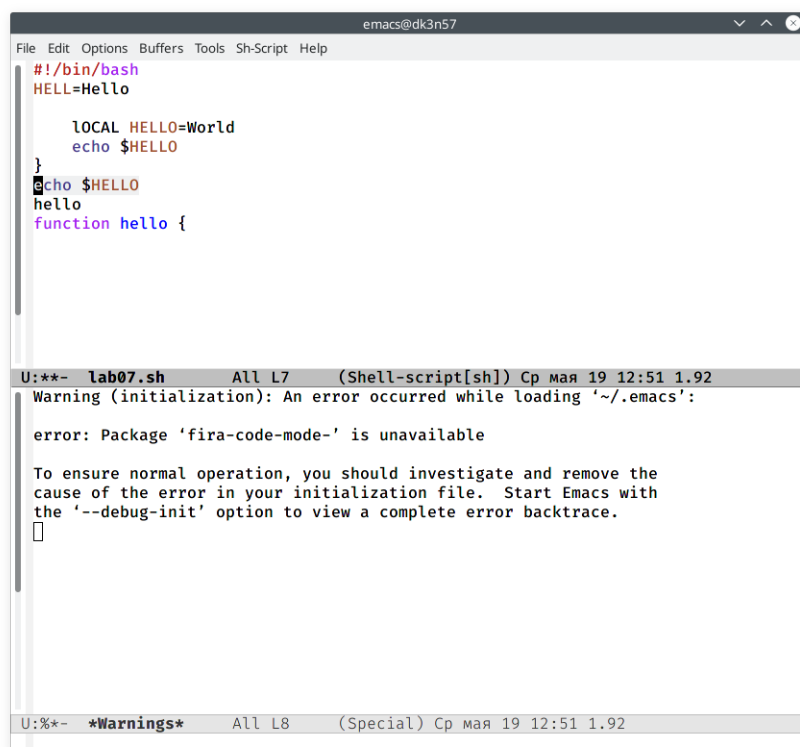


Figure 3.6: Выделила область текста

5.4. Скопировала область в буфер обмена («Alt-w»).

5.5. Вставила область в конец файла («Ctrl-y») (Скриншот 3.7).

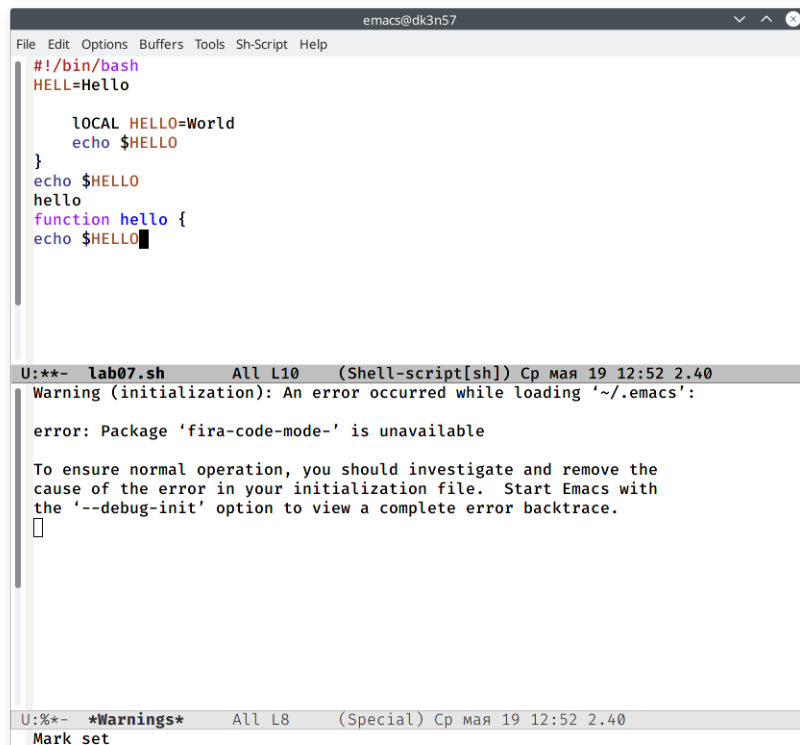
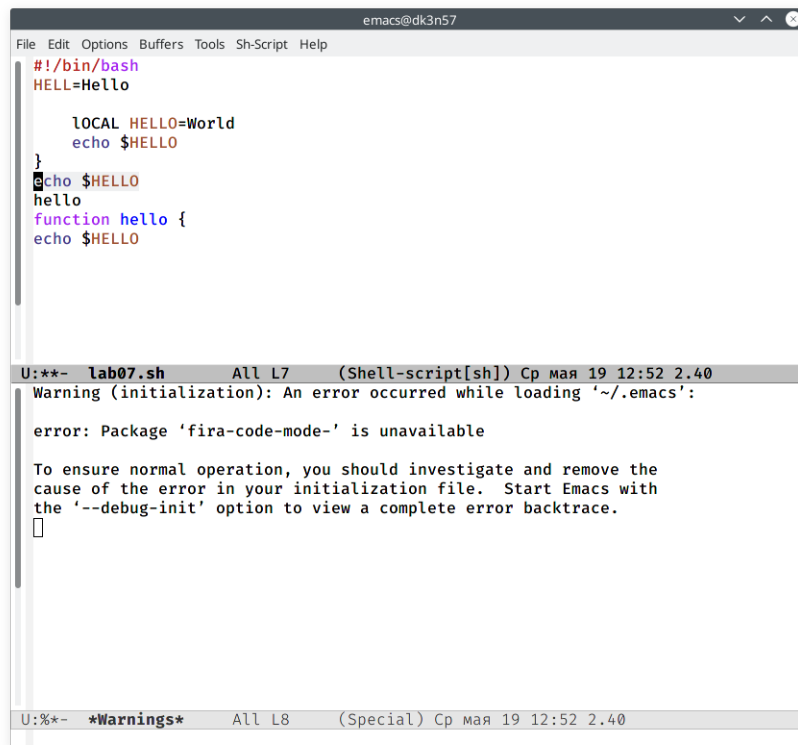


Figure 3.7: Область в конце файла

5.6. Вновь выделила эту область («Ctrl-space») (Скриншот 3.8) и на этот раз вырежем её («Ctrl-w») (Скриншот 3.9).



The screenshot shows an Emacs window titled 'emacs@dk3n57'. The main buffer contains a shell script with the following content:

```
#!/bin/bash
HELL=Hello

    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
function hello {
echo $HELLO
```

Below the script, a status bar indicates the current buffer is 'lab07.sh' and the cursor is at line 7. Below this, a message box displays the following text:

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, another status bar shows the current buffer is '*Warnings*' and the cursor is at line 8.

Figure 3.8: Выделила область



Figure 3.9: Вырезание строки

5.7. Отменила последнее действие («Ctrl-/») (См. рисунок 3.10).

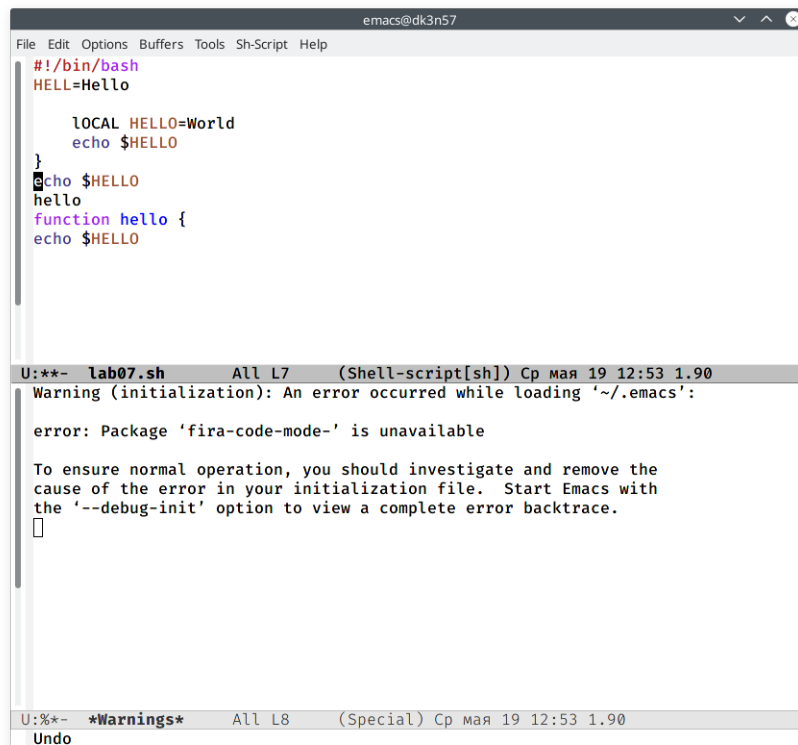
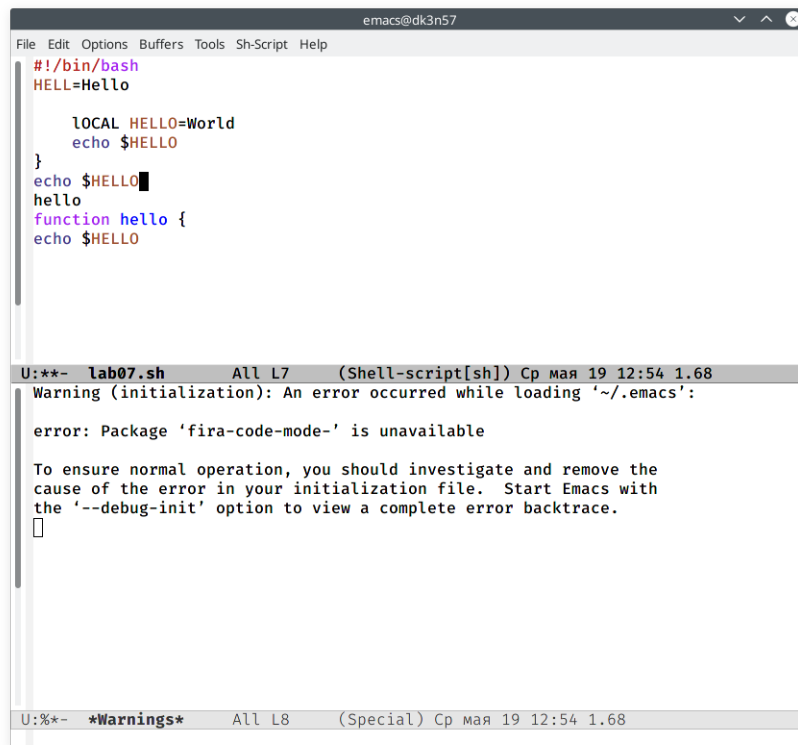


Figure 3.10: Отмена последнего действия

6). Выполним следующие действия:

6.1. Переместила курсор в начало строки («Ctrl-a») (алгоритм действий представлен на рис. 3.11 , 3.12).



The screenshot shows an Emacs window titled 'emacs@dk3n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main editing area contains a shell script with the following content:

```
#!/bin/bash
HELL=Hello

    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
function hello {
echo $HELLO
```

Below the script, a status bar indicates 'U:*** lab07.sh All L7 (Shell-script[sh]) Cp мая 19 12:54 1.68'. The buffer content shows a warning and an error:

```
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, another status bar shows 'U:%*- *Warnings* All L8 (Special) Cp мая 19 12:54 1.68'.

Figure 3.11: Переместила курсор

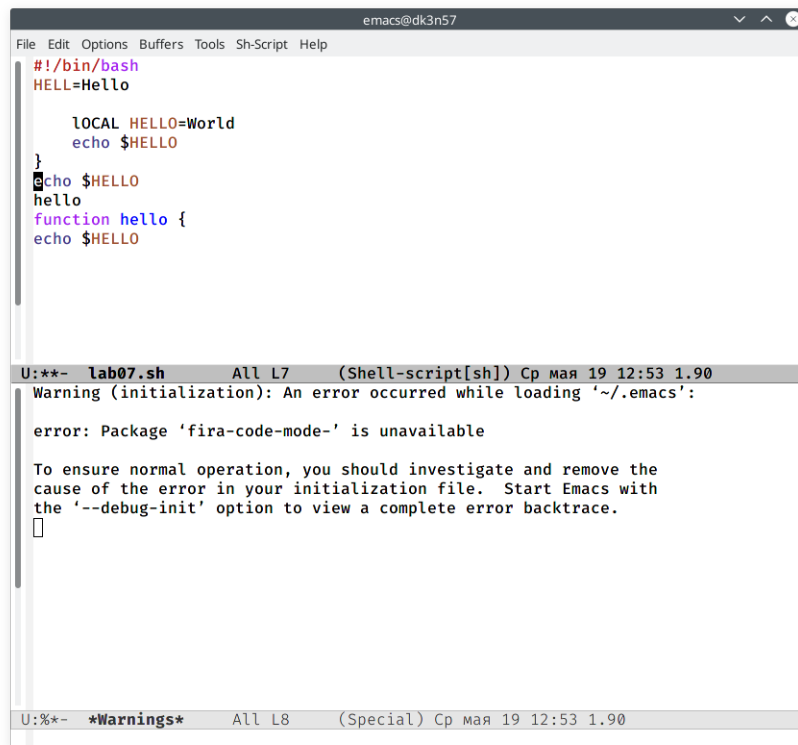


Figure 3.12: Перемещение курсора

6.2. Переместила курсор в конец строки («Ctrl-e») (Скриншот 3.13).

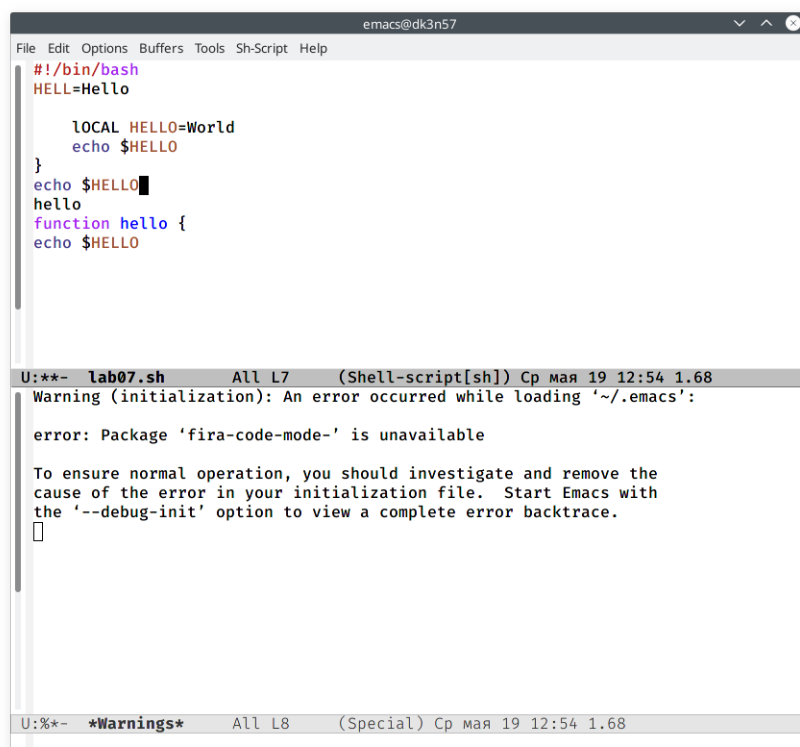


Figure 3.13: Перемещение курсора

6.3. Переместила курсор в начало буфера («Alt-<») (Скриншот 3.14).



Figure 3.14: Перемещение курсора

6.4. Переместила курсор в конец буфера («Alt->») (Скриншот 3.15).

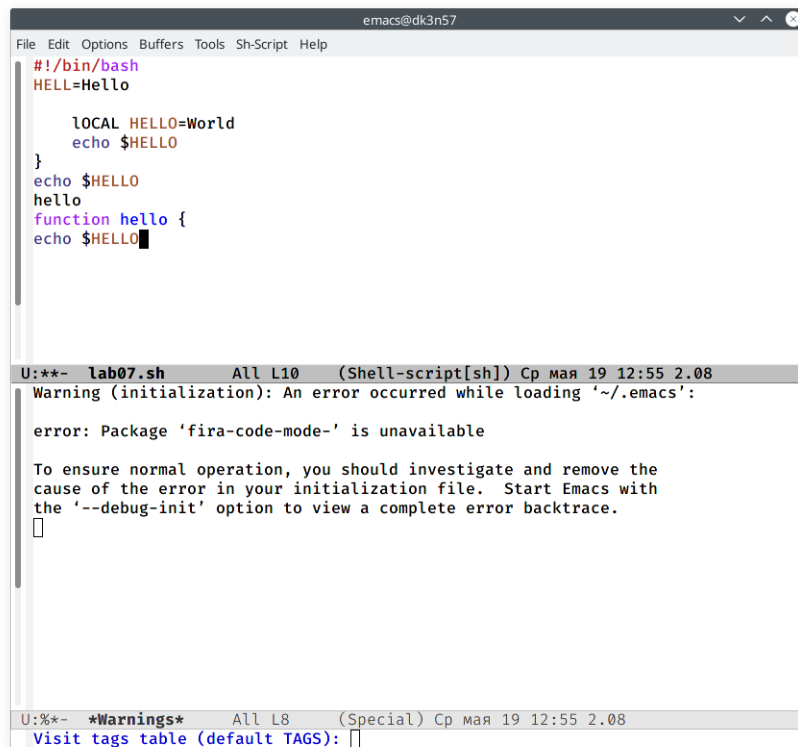


Figure 3.15: Перемещение курсора

7). Выполнила следующие действия:

7.1. Вывела список активных буферов на экран («Ctrl-x»«Ctrl-b») (алгоритм действий представлен на рис. 3.16).

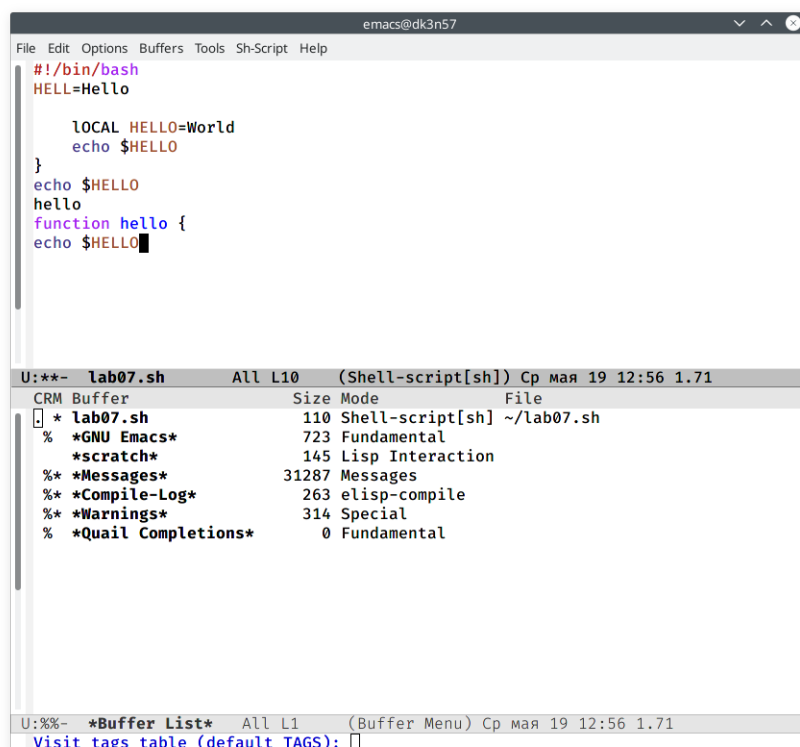


Figure 3.16: Список активных буферов

7.2. Переместилась во вновь открытое окно («Ctrl-хо») со списком открытых буферов (См. рисунок 3.17) и переключилась на другой буфер (для этого нажала на «enter» после выбора необходимого буфера) (См. рисунок 3.18).

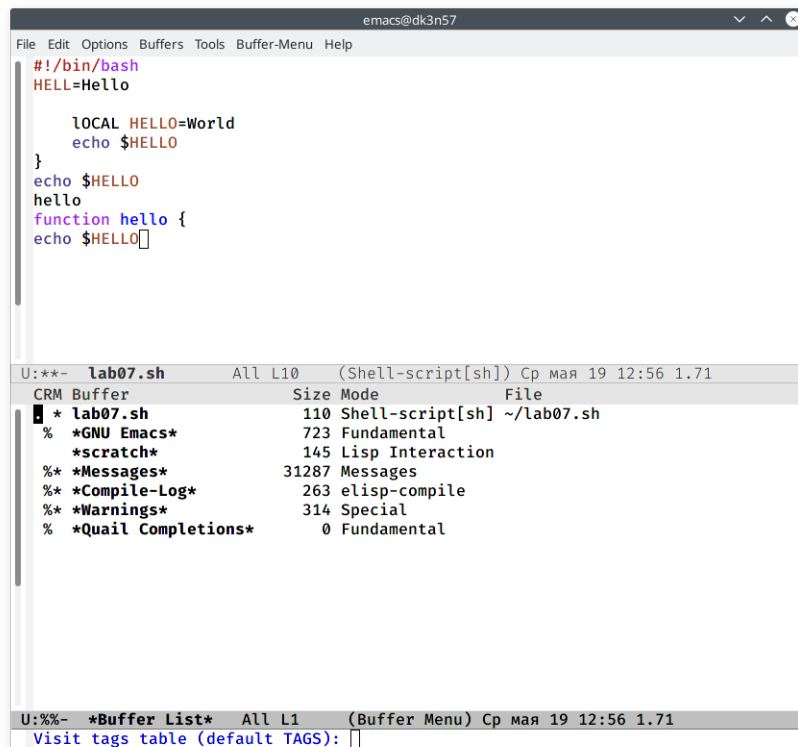
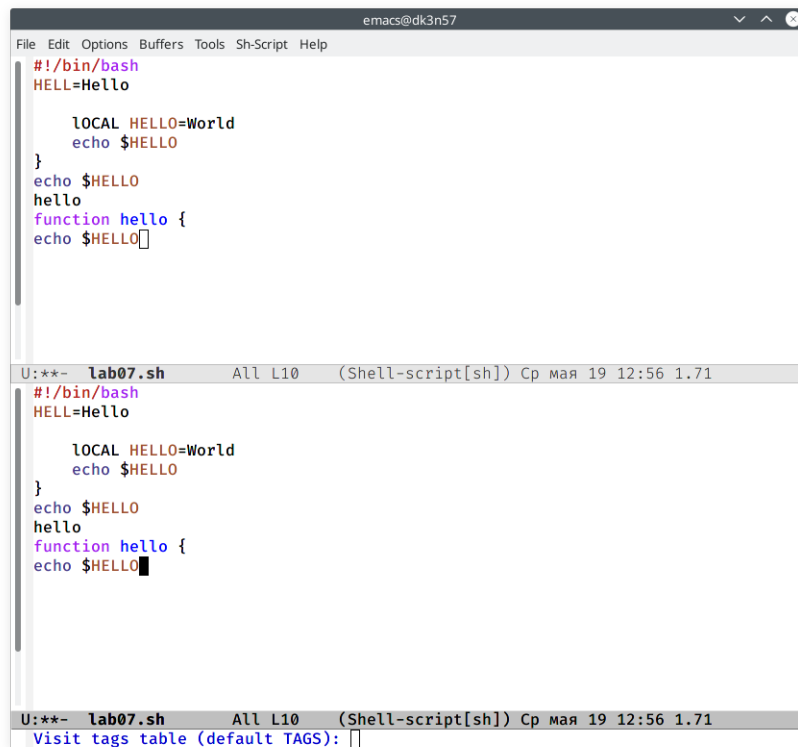


Figure 3.17: Список открытых буферов



The image shows a screenshot of the Emacs editor window titled 'emacs@dk3n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main editing area contains a shell script with the following content:

```
#!/bin/bash
HELL=Hello

    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
function hello {
echo $HELLO
```

Below the editing area, there are two buffer windows. The top buffer window is titled 'U:*-- lab07.sh All L10 (Shell-script[sh]) Ср мая 19 12:56 1.71' and contains the same shell script content as the main editing area. The bottom buffer window is titled 'U:*-- lab07.sh All L10 (Shell-script[sh]) Ср мая 19 12:56 1.71' and contains the text 'Visit tags table (default TAGS):' followed by a cursor.

Figure 3.18: Переключение на буфера

7.3. Закрывает это окно («Ctrl-x0») (алгоритм действий представлен на рис. 3.19).

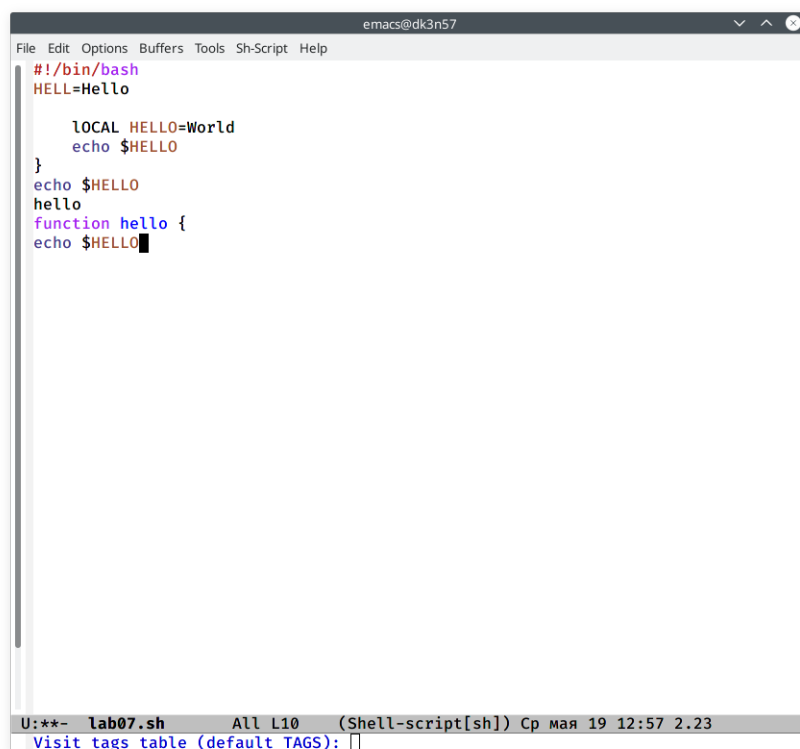


Figure 3.19: Закрывает окно буфера

7.4. Теперь вновь переключилась между буферами, но уже без вывода их списка на экран («Ctrl-x b») (алгоритм действий представлен на рис. 3.20).

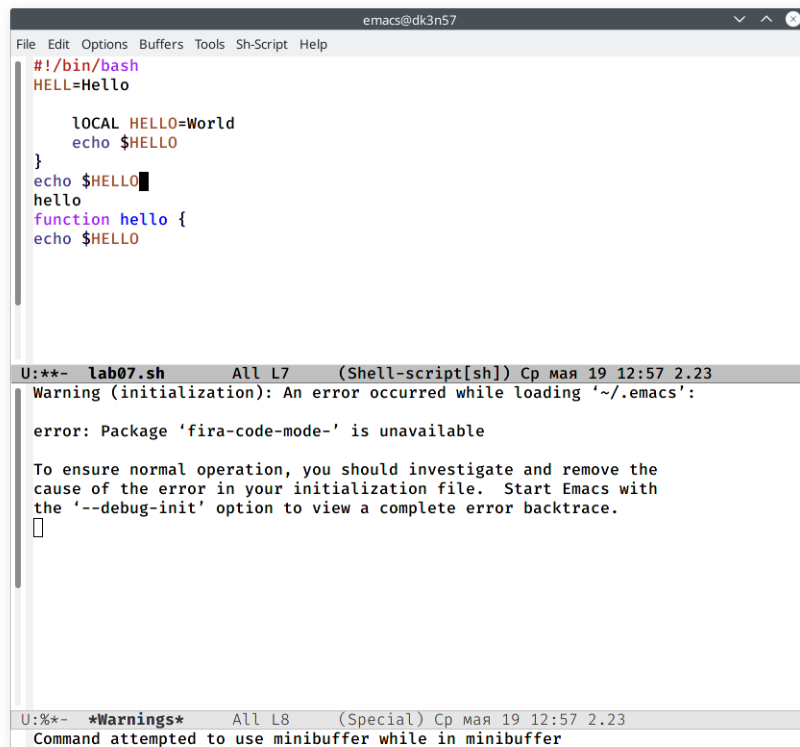


Figure 3.20: Переключение между буферами

8). Выполнила следующие действия:

8.1. Поделила фрейм на 4 части: разделила фрейм на два окна по вертикали («Ctrl-x 3»), а затем каждое из этих окон на две части по горизонтали («Ctrl-x 2») (алгоритм действий представлен на рис. 3.21 , 3.22 , 3.23).

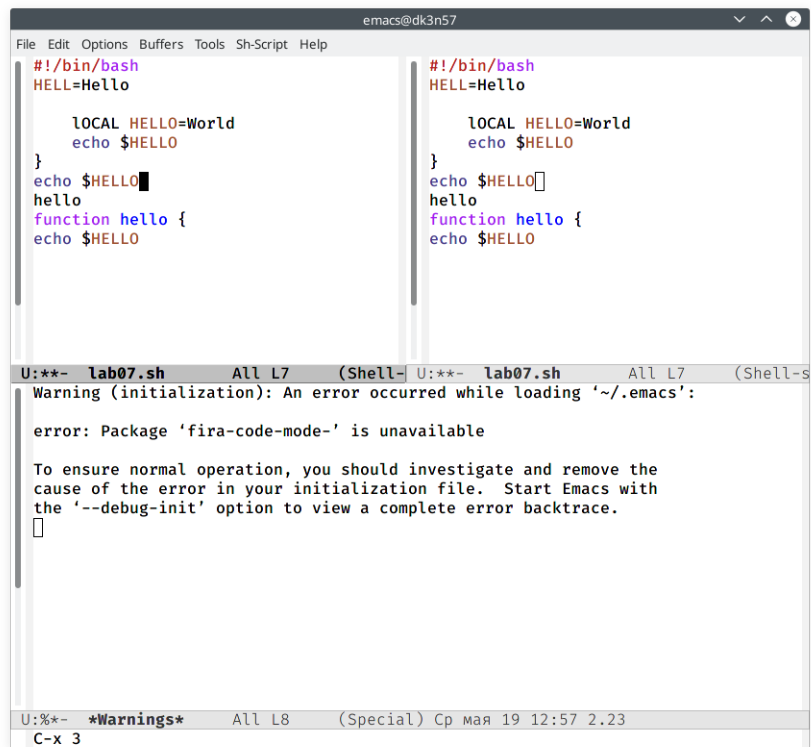


Figure 3.21: Работа с фреймом

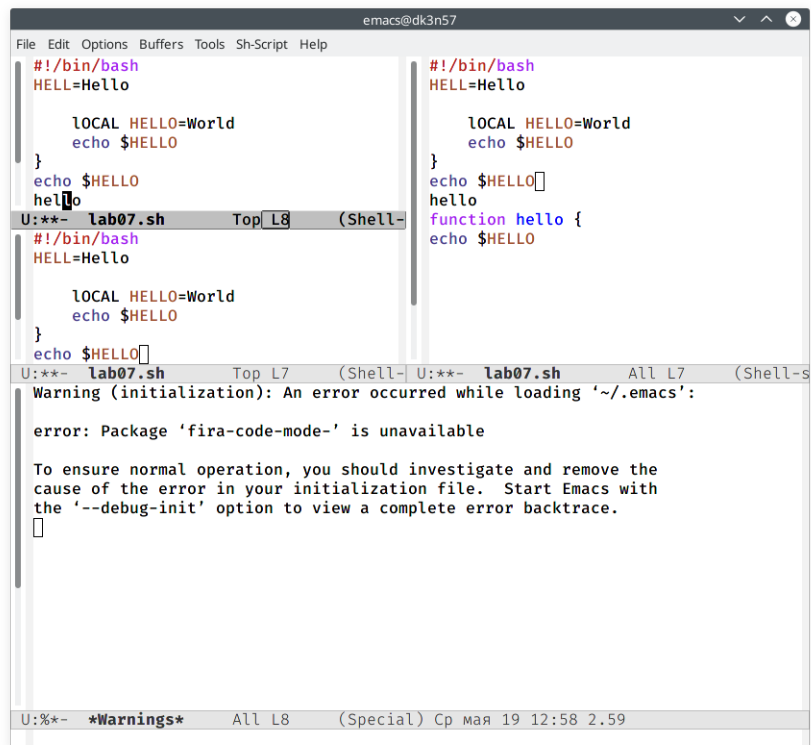


Figure 3.22: Работа с фреймом

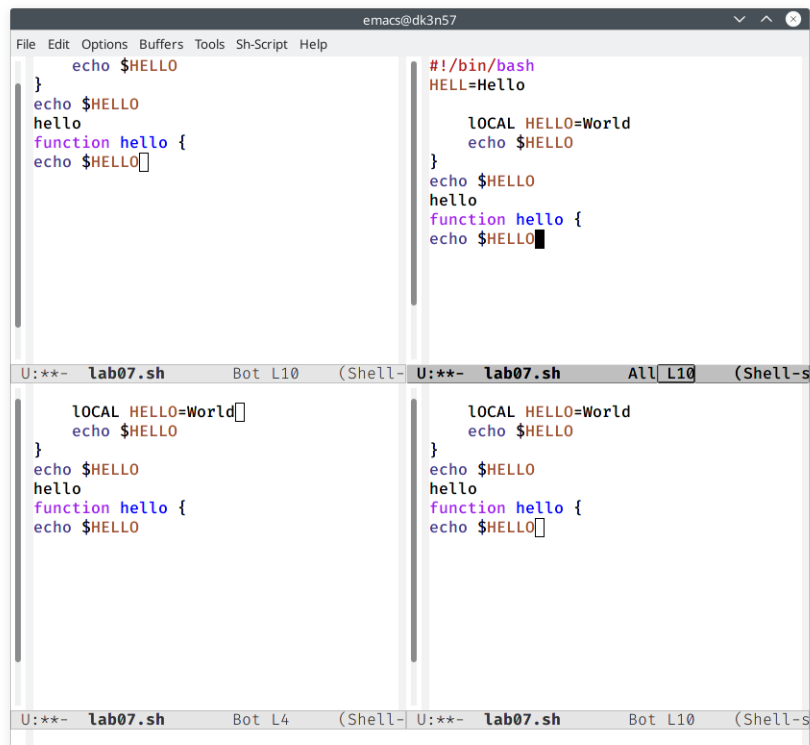


Figure 3.23: Работа с фреймом

8.2. В каждом из четырёх созданных окон открыла новый буфер (файл) и ввела несколько строк текста. Для этого предварительно создала эти файлы с помощью команд «touch example1.txt», «touch example2.txt», «touch example3.txt», «touch example4.txt» (алгоритм действий представлен на рис. 3.24 , 3.25 , 3.26).

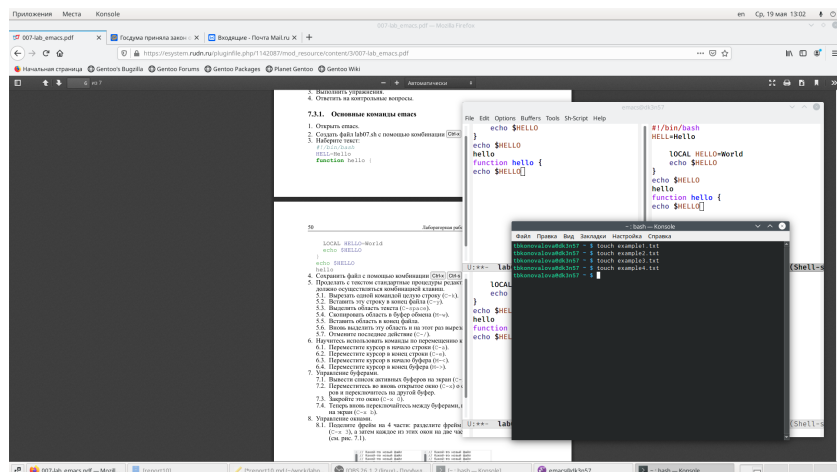


Figure 3.24: Создание файлов

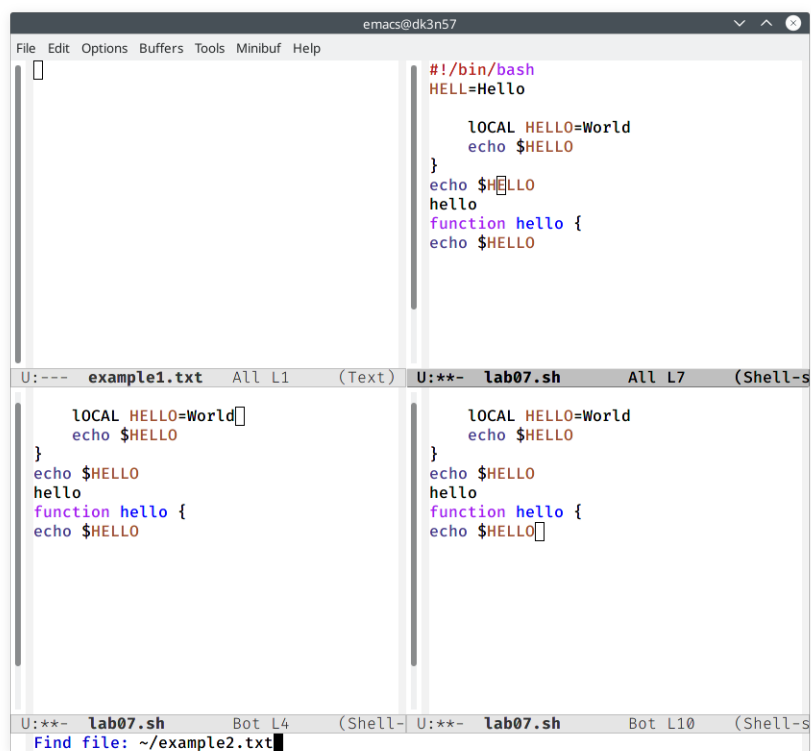


Figure 3.25: Команда для работы с файлами

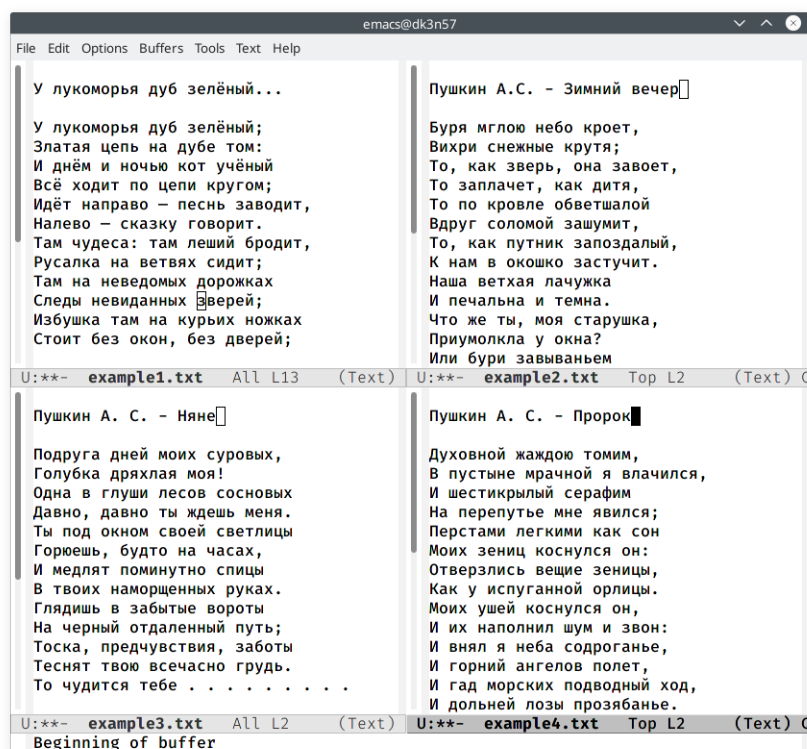


Figure 3.26: Вводим текст в документы

9). Выполнила следующие действия:

9.1. Переключила в режим поиска («Ctrl-s») и нашла несколько слов, присутствующих в тексте (алгоритм действий представлен на рис. 3.27 , 3.28).

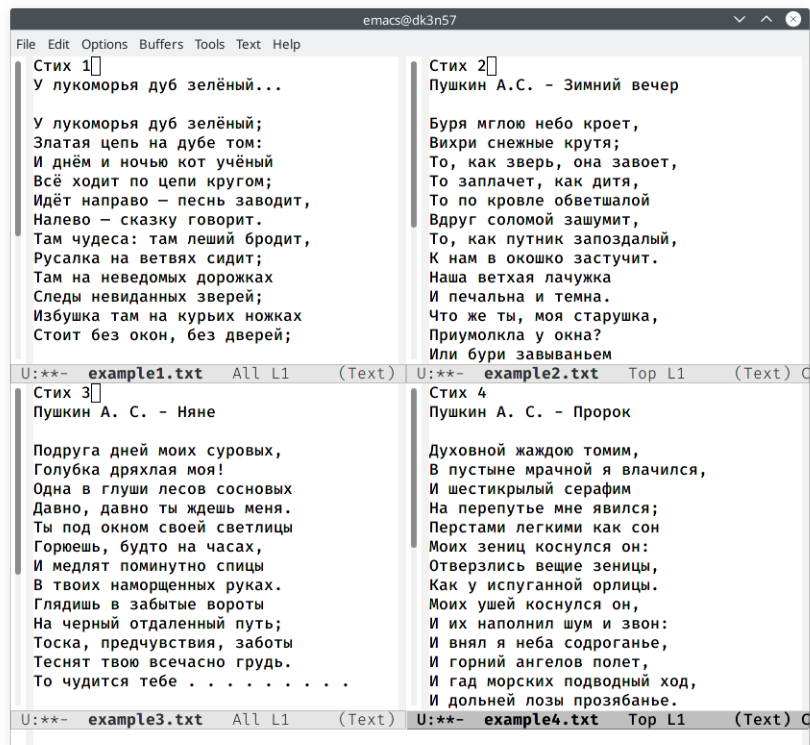


Figure 3.27: Режим поиска

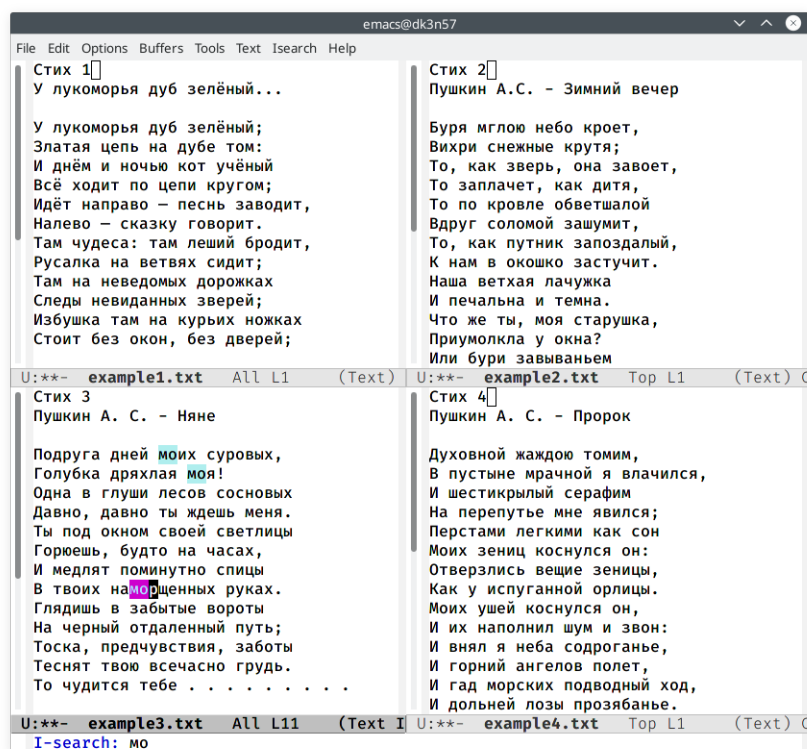


Figure 3.28: Поиск слов в тексте

9.2. Переключилась между результатами поиска, нажимая «Ctrl-s» (алгоритм действий представлен на рис. 3.29 , 3.30).

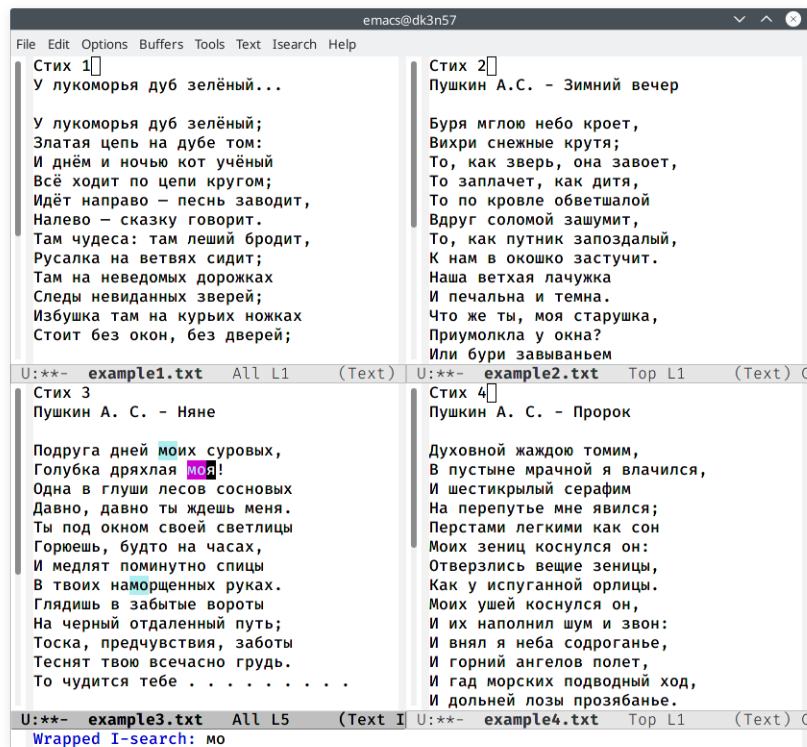


Figure 3.29: Переключение между поисками

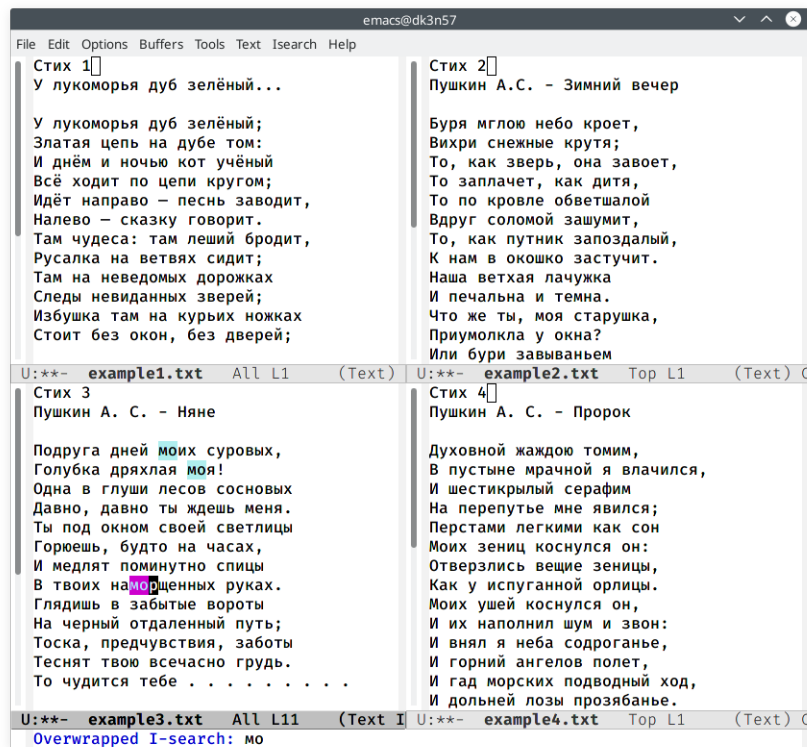


Figure 3.30: Переключение между поисками

9.3. Вышла из режима поиска, нажав «Ctrl-g» (См. рисунок 3.31)

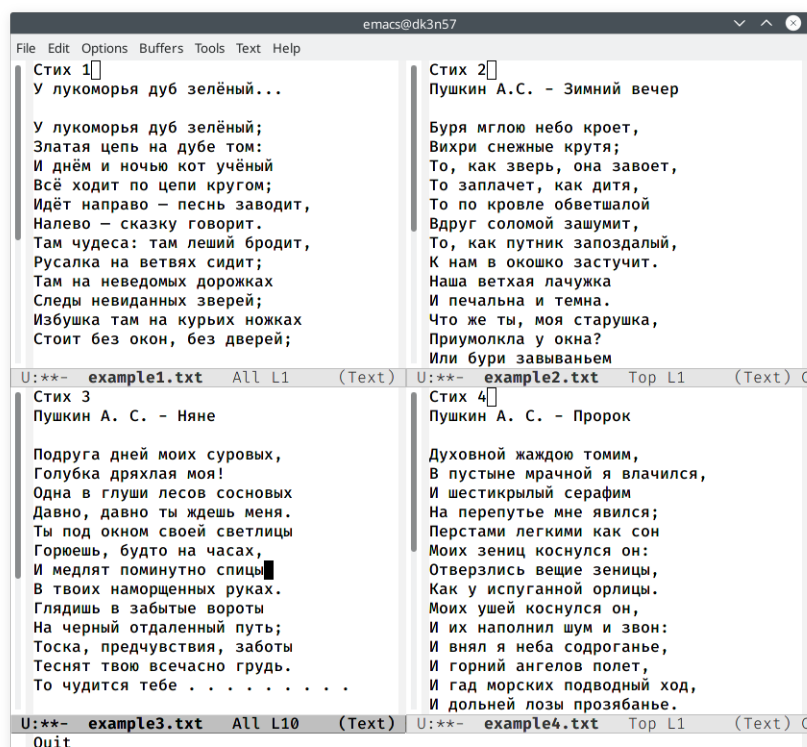


Figure 3.31: Выходим из режима поиска

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

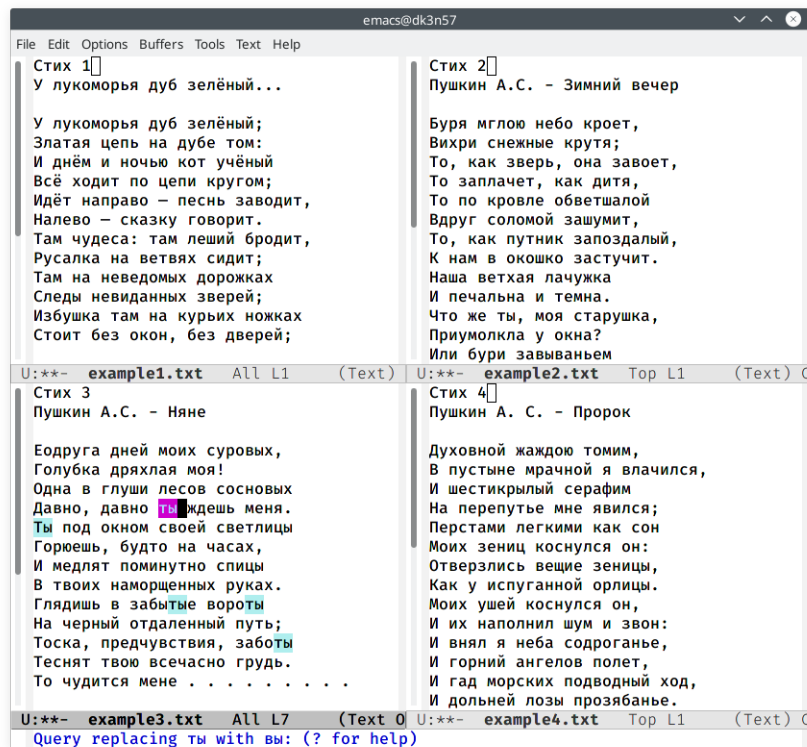


Figure 3.32: Переходим в режим поиска

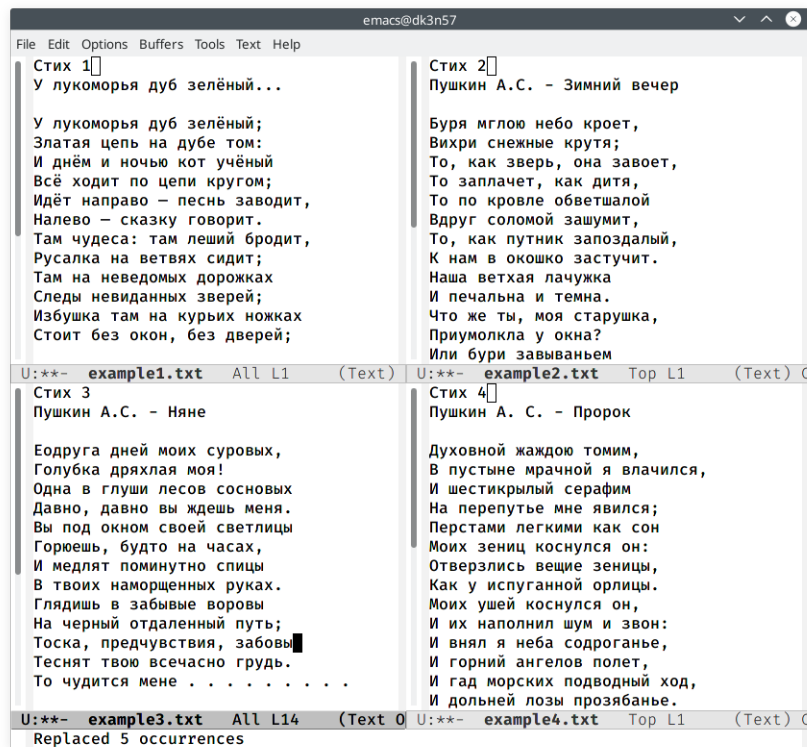


Figure 3.33: Переходим в режим поиска

9.5. Пробую другой режим поиска, нажав «Alt-so» (алгоритм действий представлен на рис. 3.34).

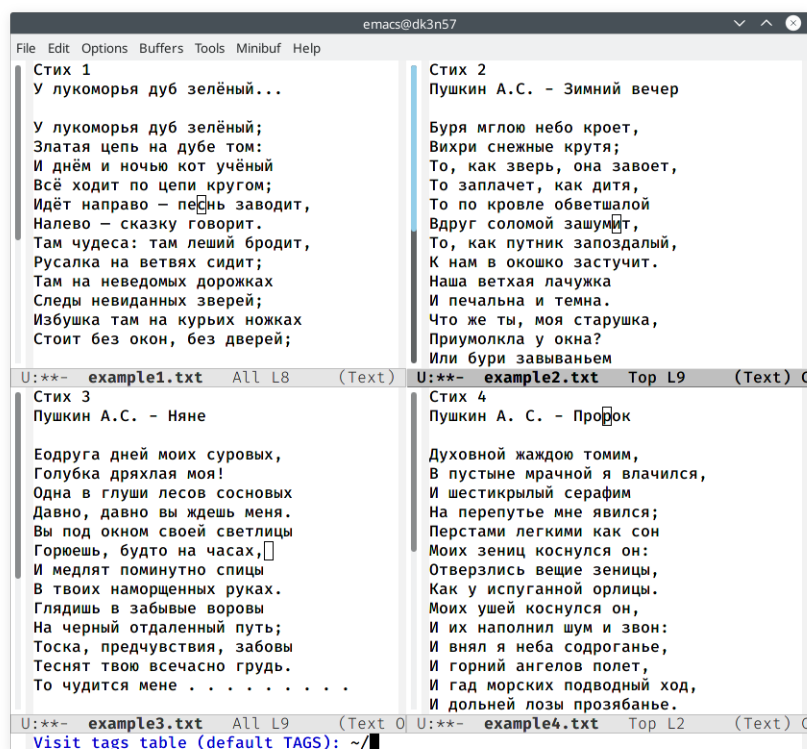


Figure 3.34: Другой режим поиска

Ответы на 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 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 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 мо-

гут 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 открыться 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 Emacs по умолчанию создаются следующие буферы: «scratch» (буфер для несохраненного текста) «Messages» (журнал ошибок, включающий также информацию, которая появляется в области EchoArea) «GNU Emacs» (справочный буфер о редакторе).

6). С-с | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и нажимаю «|» С-сC-| 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 неудобным.

4 Библиография

1. Программное обеспечение GNU/Linux. Лекция 2. Терминал и командная строка (Г. Курячий, МГУ);
2. Программное обеспечение GNU/Linux. Лекция 3. FHS и процессы (Г. Курячий, МГУ);
3. Электронный ресурс: <https://alexott.net/ru/writings/altlinux-emacs/>
4. Электронный ресурс: <https://yandex.ru/turbo/pythonist.ru/s/tekstovyj-redaktor-emacs/>

5 Выводы

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