

Отчет по лабораторной работе №10

Дисциплина: Операционные системы

Калистратова Ксения Евгеньевна

Содержание

Цель работы	1
Задачи.....	1
Выполнение лабораторной работы	1
Контрольные вопросы	18
Выводы	19
Библиография.....	19

Цель работы

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

Задачи

1. Познакомиться с текстовым редактором emacs.
2. Изучить команды управления (для перемещения курсора, работы с текстом, работы с выделенной областью текста, для поиска и замены).
3. В ходе работы использовать эти команды и интерпретировать их вывод.
4. Выполнить отчет.

Выполнение лабораторной работы

1. Откроем редактор Emacs с помощью команды «emacs &». (рис. 1)

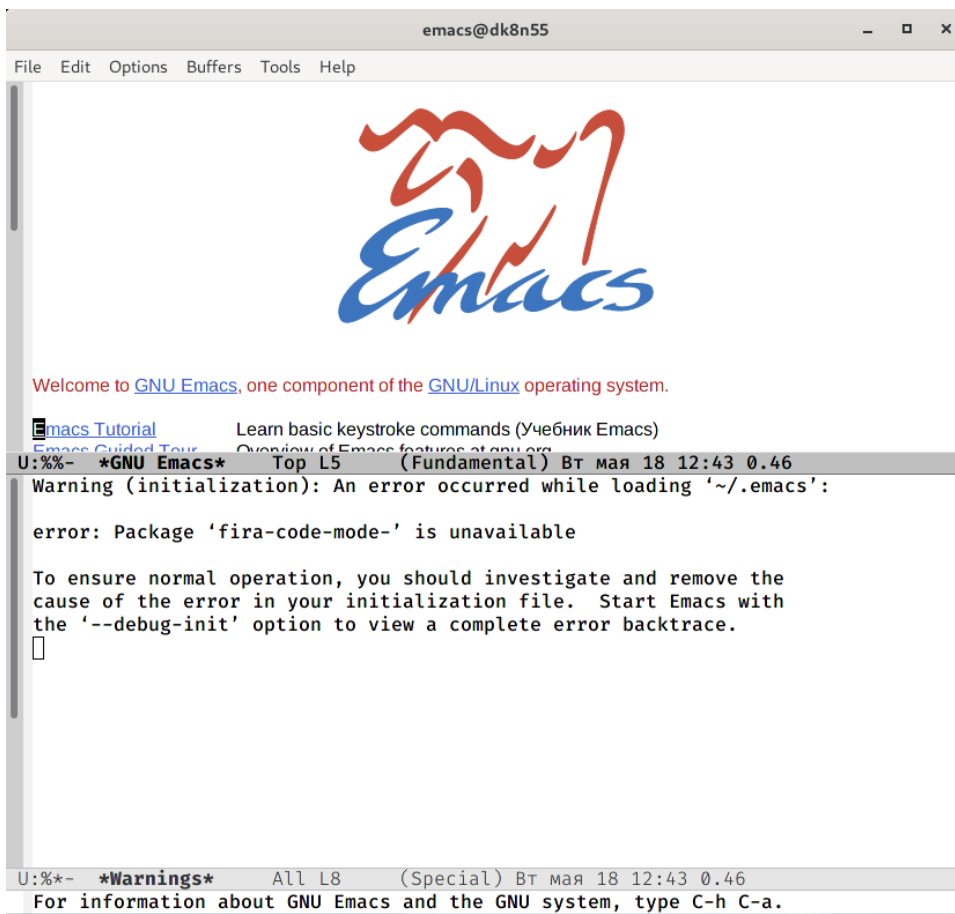


Figure 1: Текстовый редактор emacs

2. Создадим файл lab07.sh с помощью комбинации «Ctrl-x»«Ctrl-f». (рис. 2)

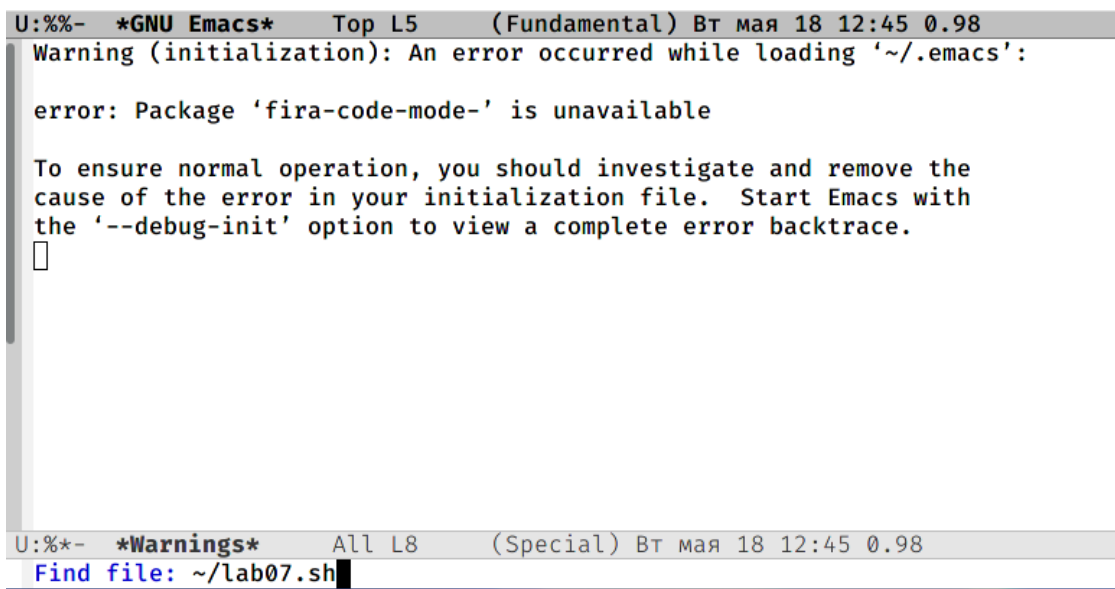
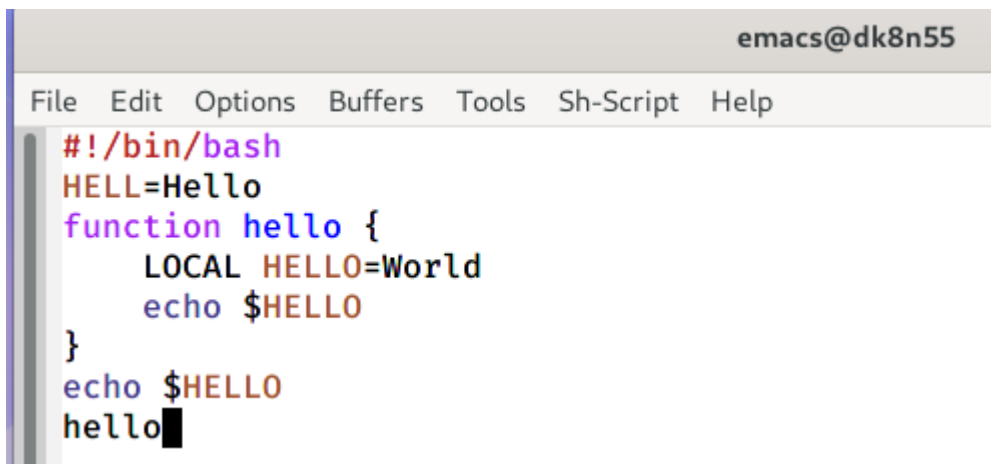


Figure 2: Создание файла lab07.sh

3. В открывшемся буфере наберем необходимый текст. (рис. 3)

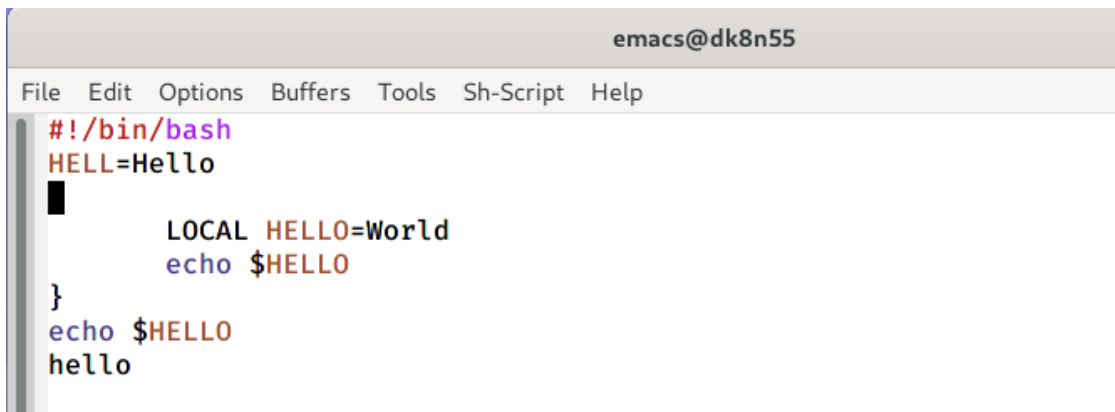


The screenshot shows the Emacs editor window with the title 'emacs@dk8n55'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The text being typed is a shell script: `#!/bin/bash`, `HELL=Hello`, `function hello {`, `LOCAL HELLO=World`, `echo $HELLO`, `}`, `echo $HELLO`, and `hello`. The cursor is at the end of the last line.

Figure 3: Ввод текста

4. Сохраним файл с помощью комбинации «Ctrl-x»«Ctrl-s».
5. Прделаем с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш.

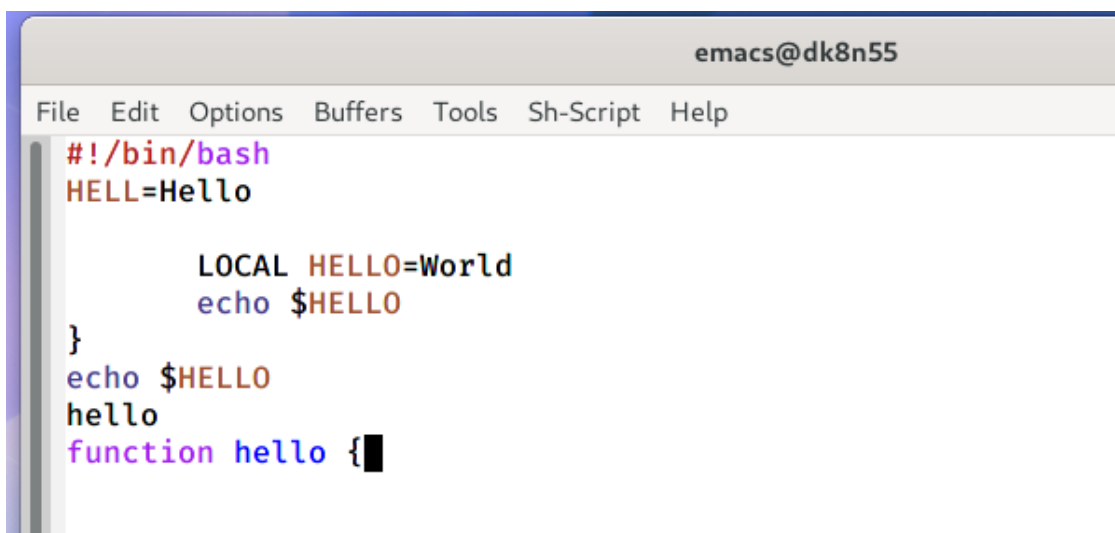
5.1. Вырежем одной командой целую строку («Ctrl-k»). (рис. 4)



The screenshot shows the Emacs editor window with the title 'emacs@dk8n55'. The menu bar is the same. The text is the same as in Figure 3, but the line `HELL=Hello` has been cut, leaving a blank line in its place. The cursor is at the end of the blank line.

Figure 4: Вырезаем одну из строк

5.2. Вставим эту строку в конец файла («Ctrl-y»). (рис. 5)



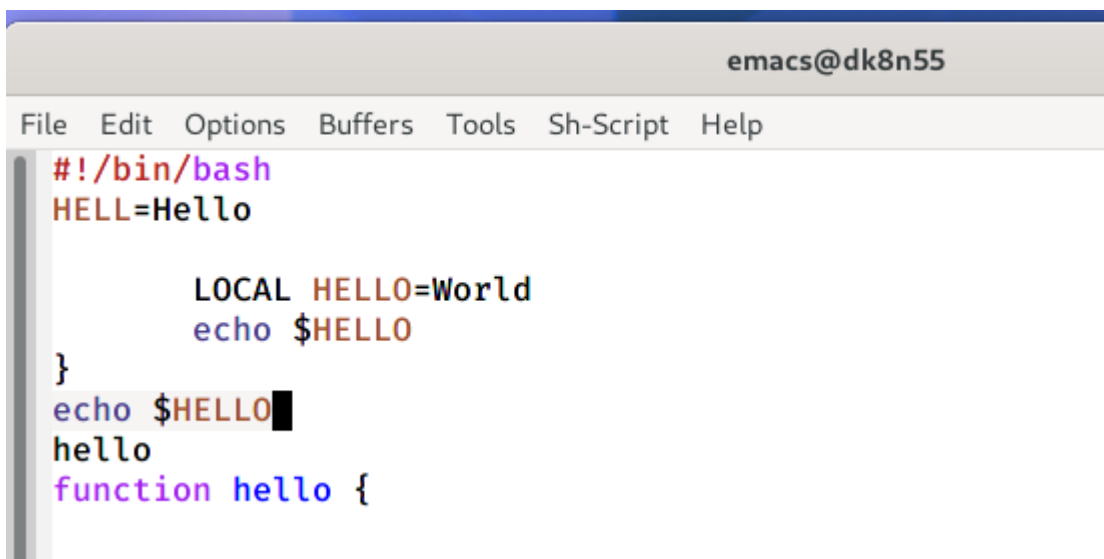
The screenshot shows the Emacs editor window titled 'emacs@dk8n55'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The code in the buffer is a shell script: `#!/bin/bash`, `HELL=Hello`, a blank line, `LOCAL HELLO=World`, `echo $HELLO`, `}`, `echo $HELLO`, `hello`, and `function hello {`. The cursor is at the end of the last line.

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

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

Figure 5: Вставка строки в конец

5.3. Выделим область текста («Ctrl-space»). (рис. 6)



The screenshot shows the same Emacs editor window. The line `echo $HELLO` is now highlighted with a light gray background, indicating it has been selected using the 'Ctrl-space' command. The rest of the script remains the same.

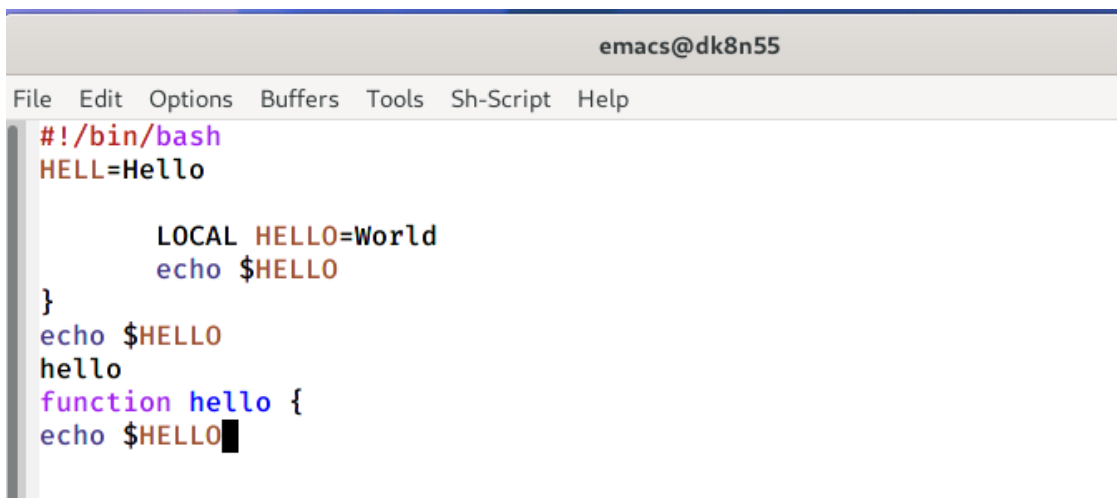
```
#!/bin/bash
HELL=Hello

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

Figure 6: Выделяем текст

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

5.5. Вставим область в конец файла («Ctrl-y»). (рис. 7)

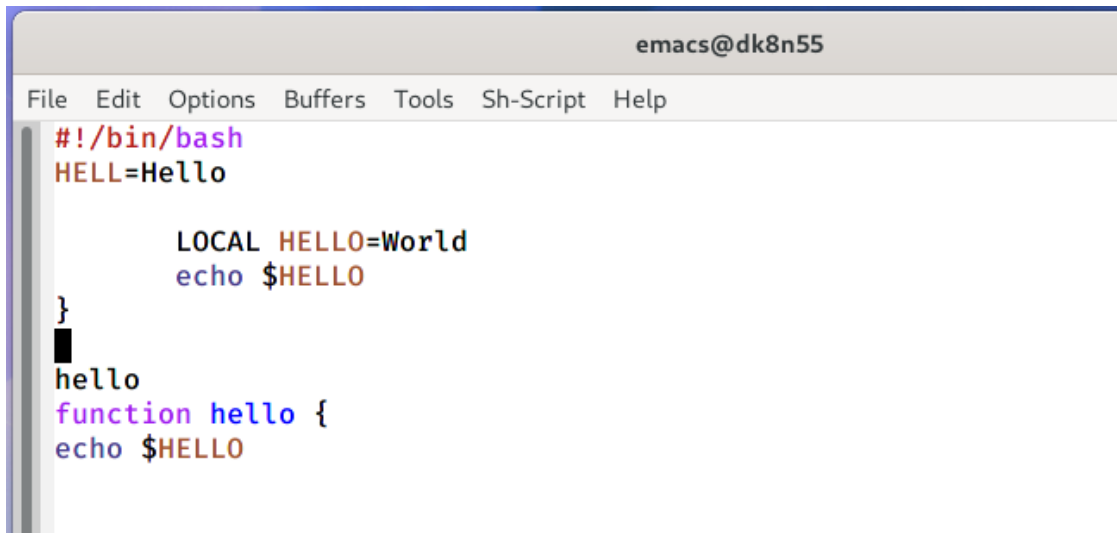


```
emacs@dk8n55
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello

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

Figure 7: Вставка текста в конец

5.6. Вновь выделим эту область («Ctrl-space») и на этот раз вырежем её («Ctrl-w»). (рис. 8)

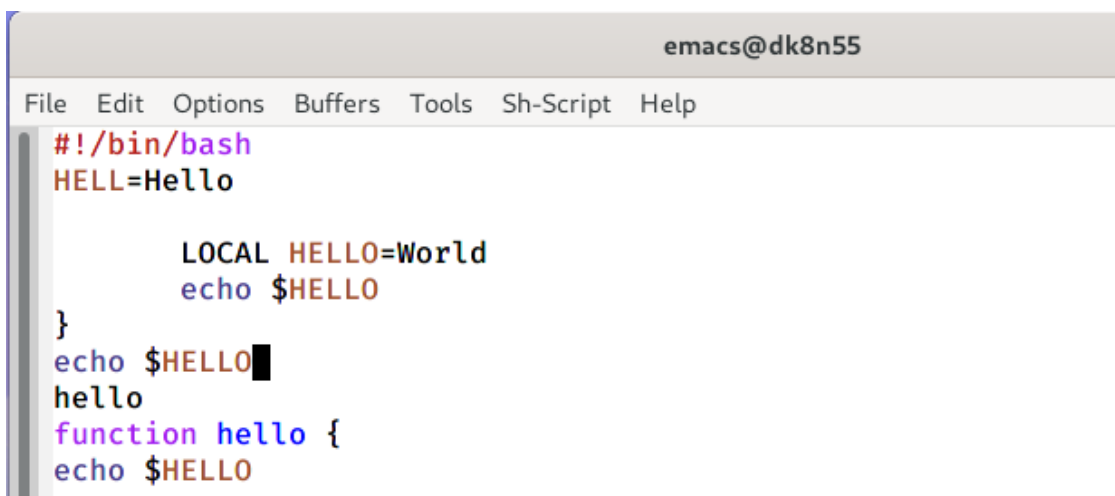


```
emacs@dk8n55
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello

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

Figure 8: Вырезаем эту область текста

5.7. Отменим последнее действие («Ctrl-/»). (рис. 9)



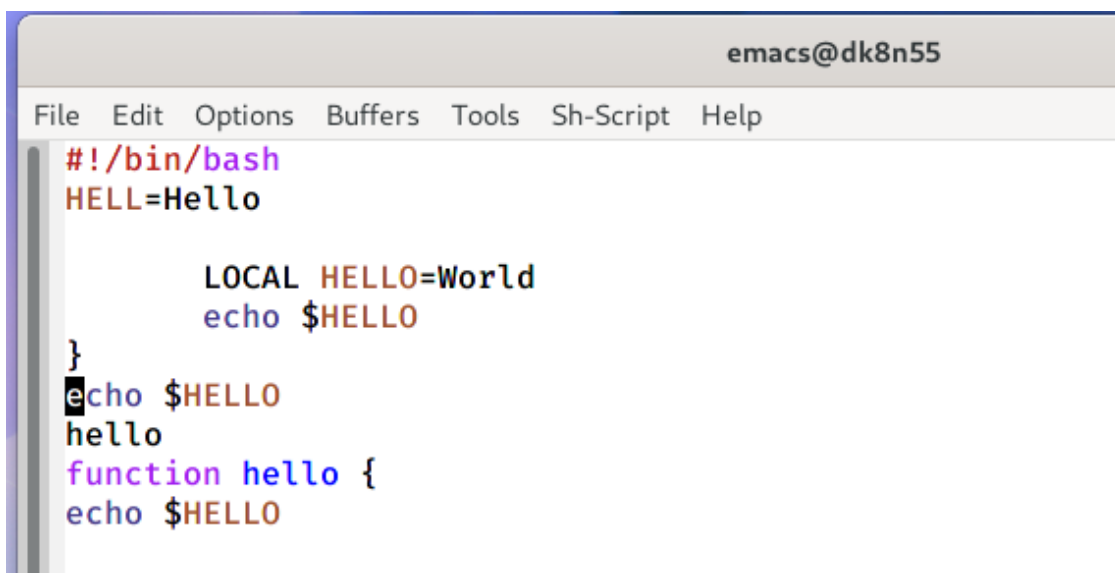
```
emacs@dk8n55
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello

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

Figure 9: Отмена последнего действия

6. Учимся использовать команды по перемещению курсора.

6.1. Переместим курсор в начало строки («Ctrl-a»). (рис. 10)

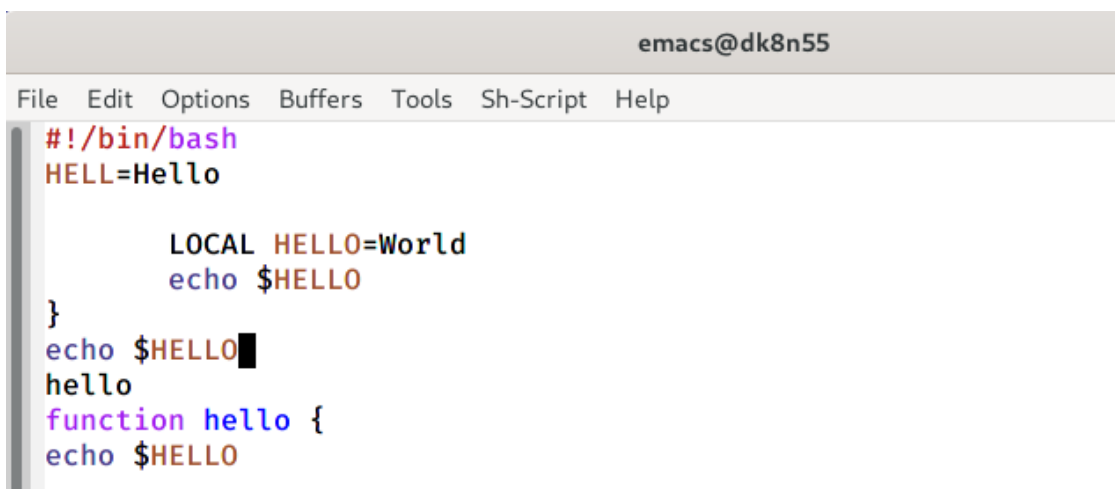


```
emacs@dk8n55
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello

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

Figure 10: Курсор в начале строки

6.2. Переместим курсор в конец строки («Ctrl-e»). (рис. 11)



The screenshot shows the Emacs editor window titled "emacs@dk8n55". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". The editor contains a shell script with the following content:

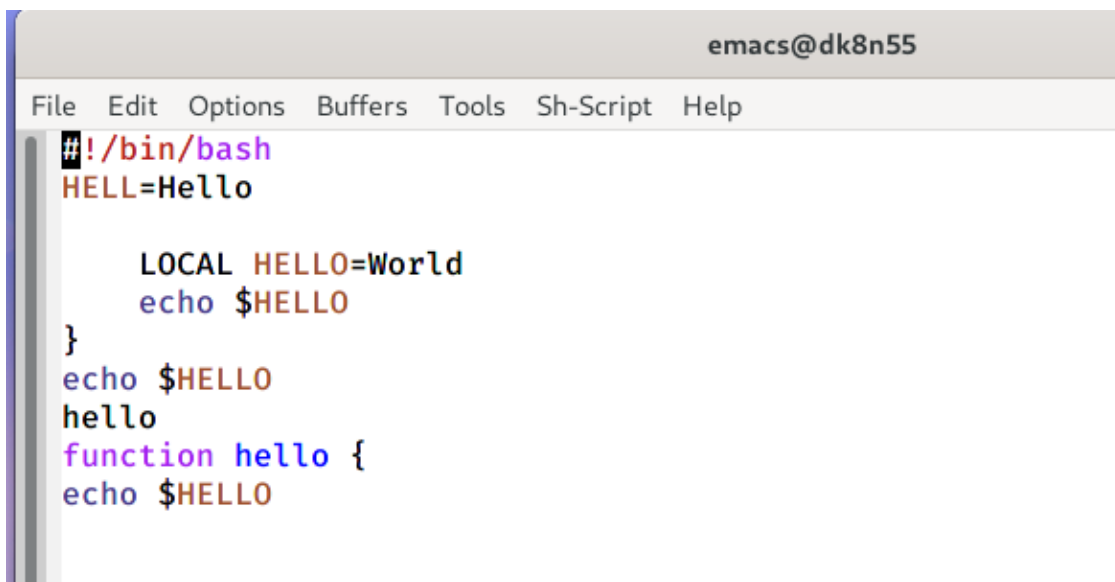
```
#!/bin/bash
HELL=Hello

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

The cursor is positioned at the end of the first line, after the text "HELL=Hello".

Figure 11: Курсор в конце строки

6.3. Переместим курсор в начало буфера («Alt-<») (рис. 12)



The screenshot shows the Emacs editor window titled "emacs@dk8n55". The menu bar is the same as in Figure 11. The editor contains the same shell script:

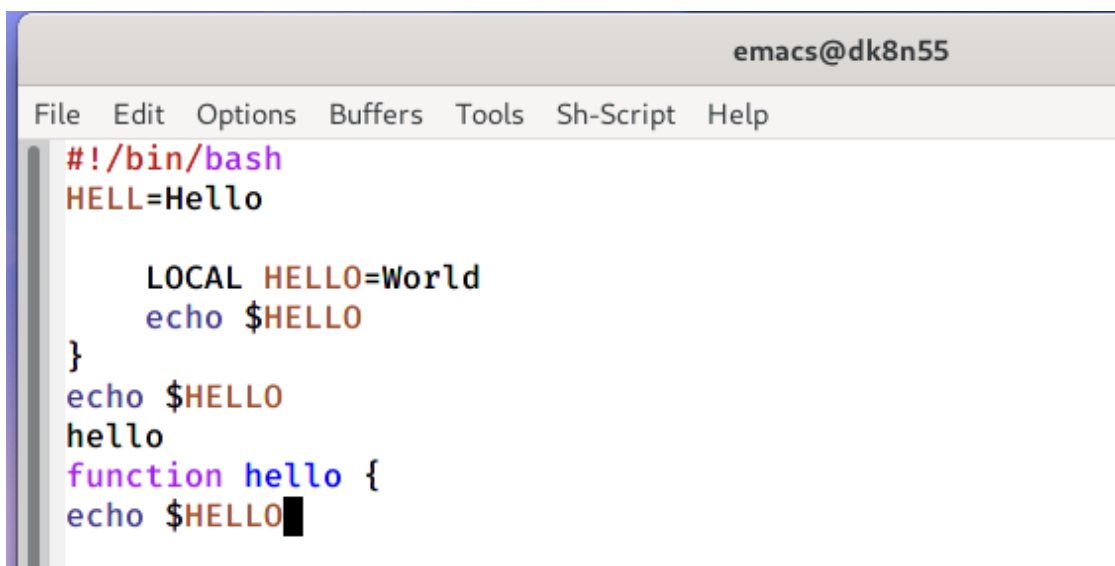
```
#!/bin/bash
HELL=Hello

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

The cursor is now positioned at the beginning of the first line, before the text "#!/bin/bash".

Figure 12: Курсор в начало буфера

6.4. Переместим курсор в начало буфера («Alt->»). (рис. 13)



```
emacs@dk8n55
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello

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

Figure 13: Курсор в конец буфера

7. Управление буферами.

7.1. Выведем список активных буферов на экран («Ctrl-x»«Ctrl-b»). (рис. 14)

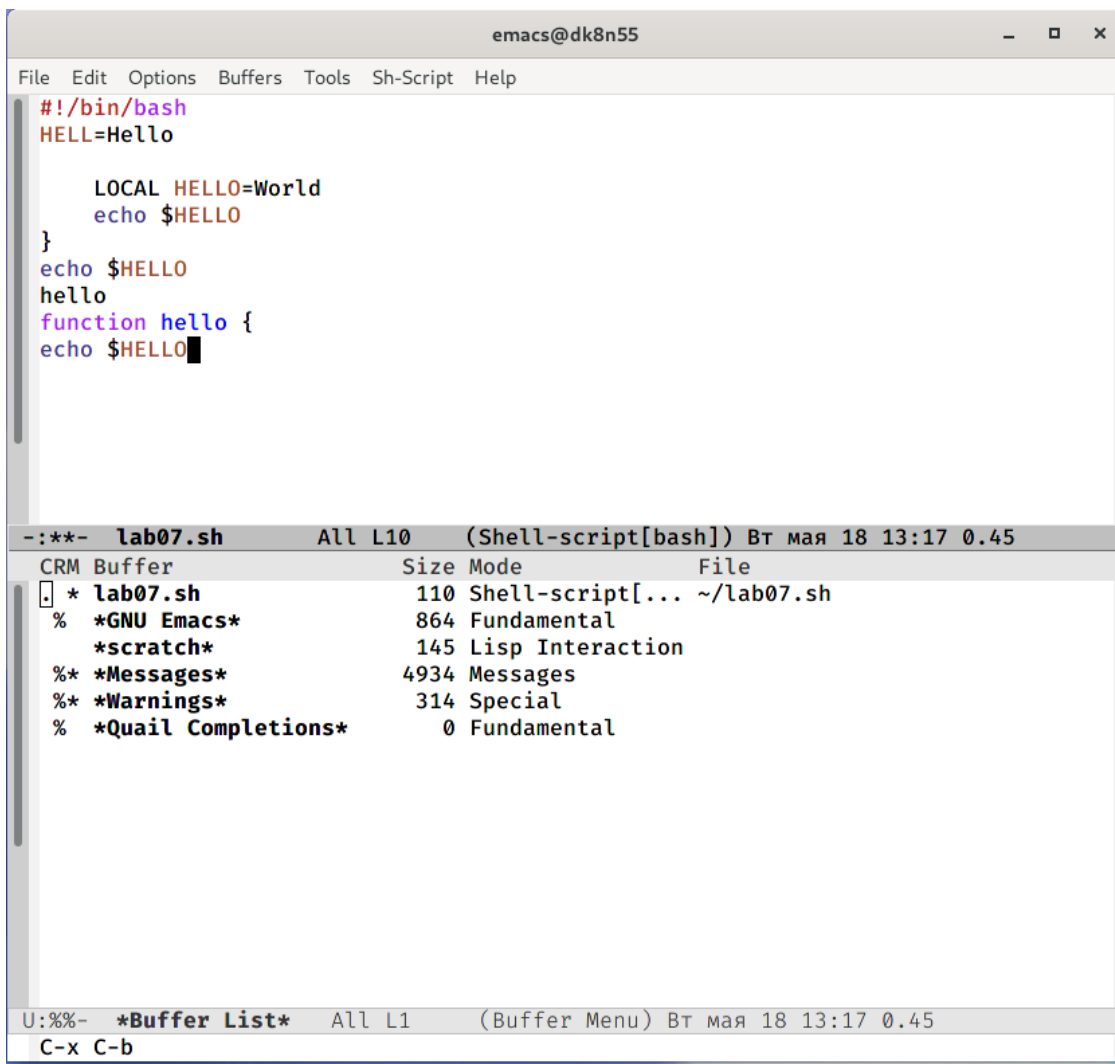


Figure 14: Список активных буферов

7.2. Переместимся во вновь открытое окно («Ctrl-хо») со списком открытых буферов и переключимся на другой буфер (для этого необходимо нажать на «enter» после выбора необходимого буфера). (рис. 15)

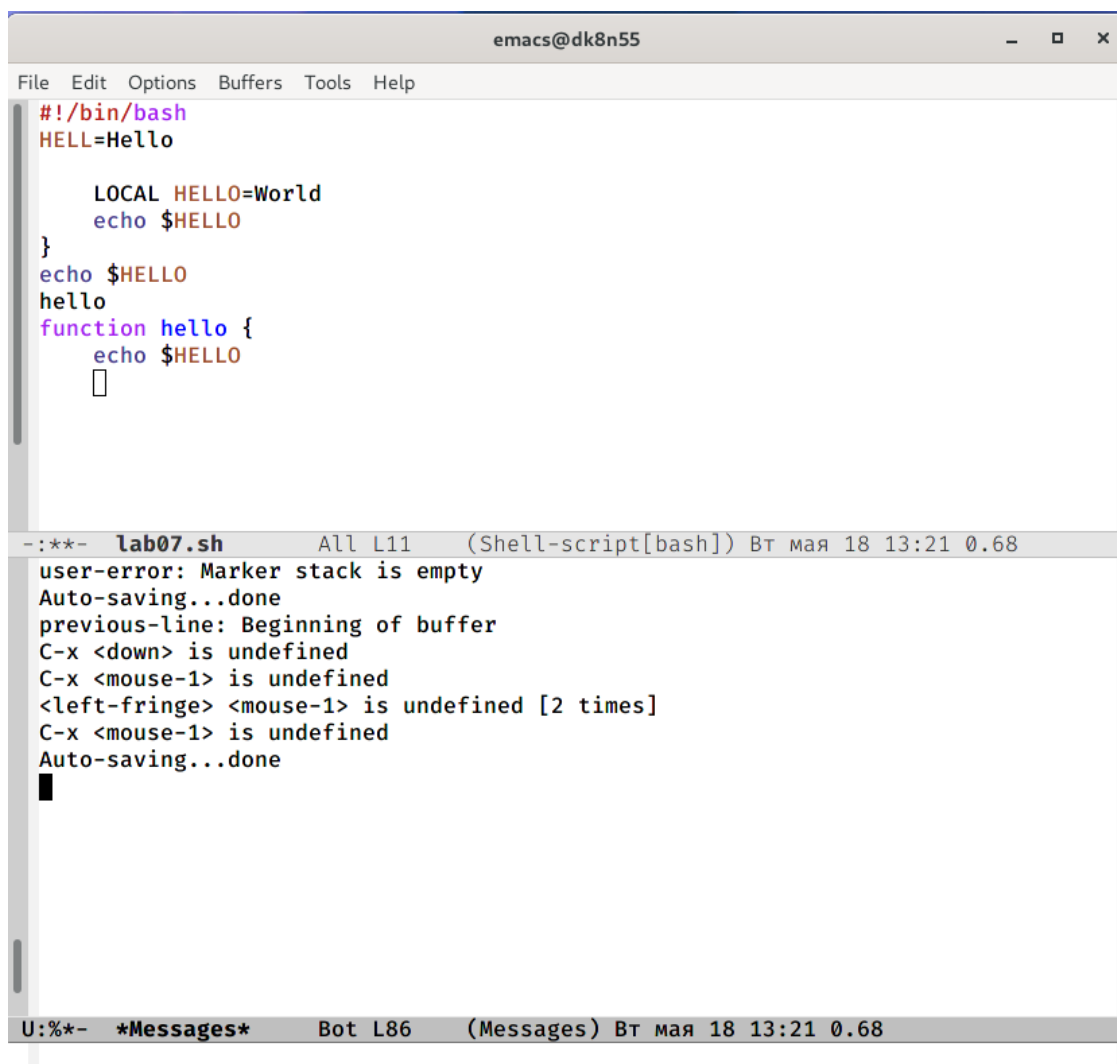
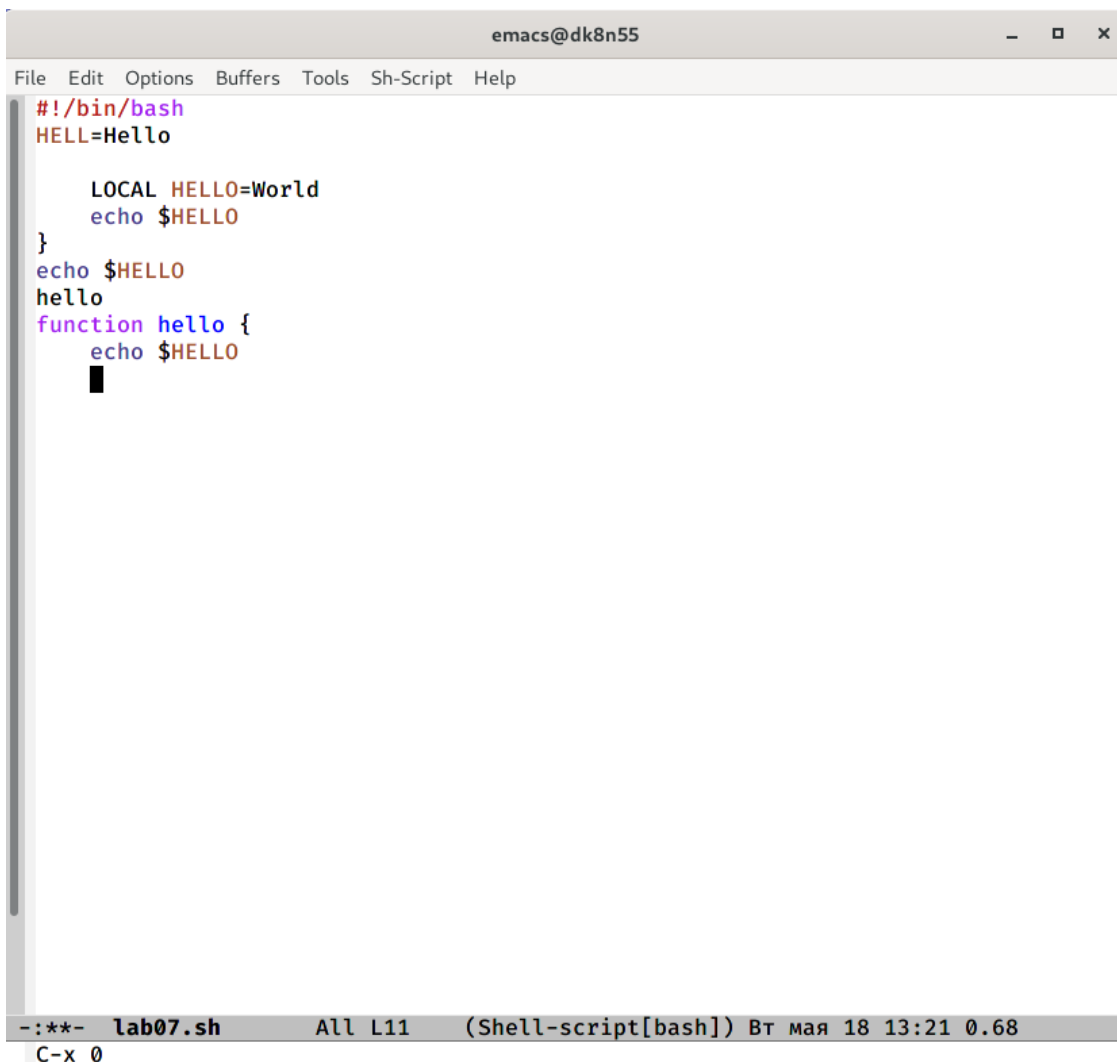


Figure 15: Переключение на другой буфер

7.3. Закроем это окно. (рис. 16)



The screenshot shows the Emacs editor window titled 'emacs@dk8n55'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main text area contains a shell script with the following content:

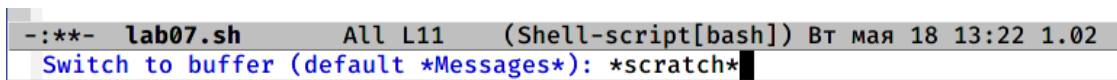
```
#!/bin/bash
HELL=Hello

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

The status bar at the bottom displays: '-:***- lab07.sh All L11 (Shell-script[bash]) Вт мая 18 13:21 0.68 C-x 0'.

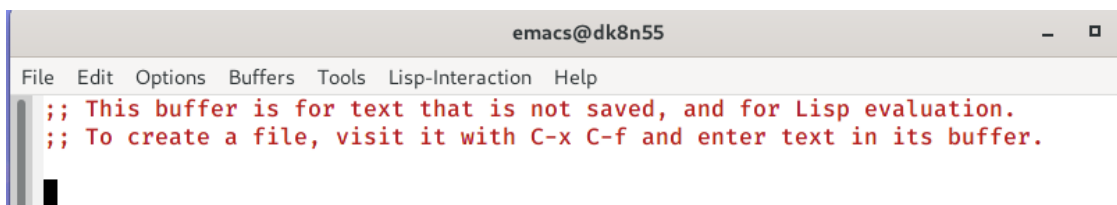
Figure 16: Заккрытие окна

7.4. Теперь вновь переключимся между буферами, но уже без вывода их списка на экран («Ctrl-x b»). (рис. 17) (рис. 18)



The screenshot shows the Emacs editor window titled 'emacs@dk8n55'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Lisp-Interaction', and 'Help'. The status bar at the bottom displays: '-:***- lab07.sh All L11 (Shell-script[bash]) Вт мая 18 13:22 1.02'. Below the status bar, the text 'Switch to buffer (default *Messages*): *scratch*' is visible, indicating the transition to the scratch buffer.

Figure 17: Переход к другому буферу



The screenshot shows the Emacs editor window titled 'emacs@dk8n55'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Lisp-Interaction', and 'Help'. The main text area contains the following text:

```
;; This buffer is for text that is not saved, and for Lisp evaluation.
;; To create a file, visit it with C-x C-f and enter text in its buffer.
```

The status bar at the bottom is empty.

Figure 18: Буфер scratch

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. 19)



Figure 19: Деление фрейма на 4 части

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. 20) (рис. 21) (рис. 22)

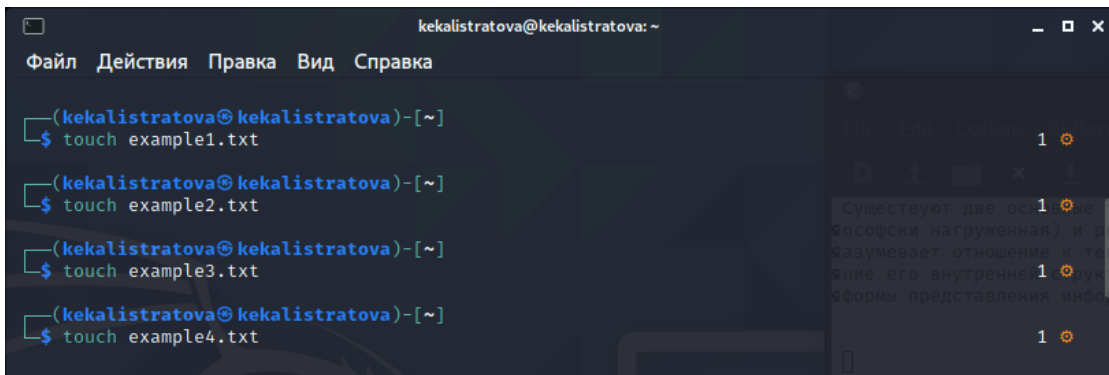


Figure 20: Создание 4 файлов

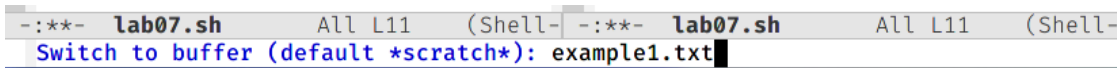


Figure 21: Переход к новому буферу

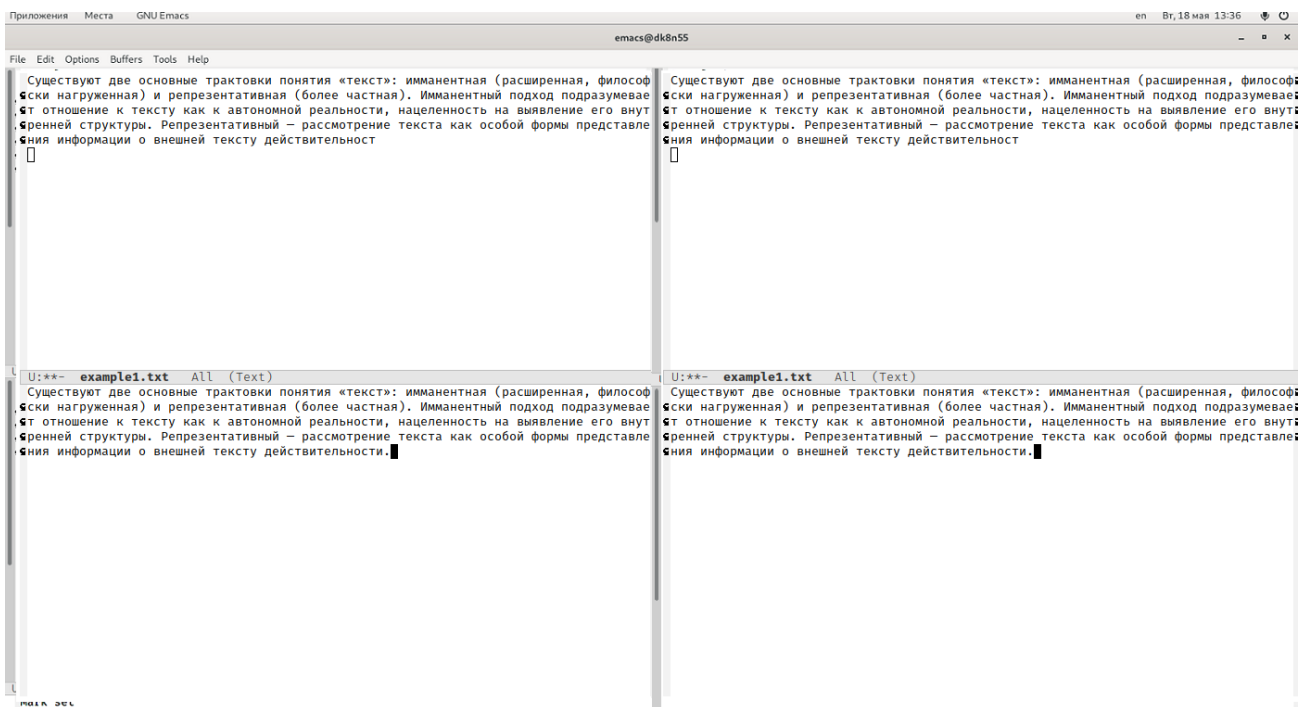


Figure 22: Новые буферы

9. Режим поиска.

9.1. Переключимся в режим поиска («Ctrl-s») и найдем несколько слов, присутствующих в тексте. (рис. 23) (рис. 24)

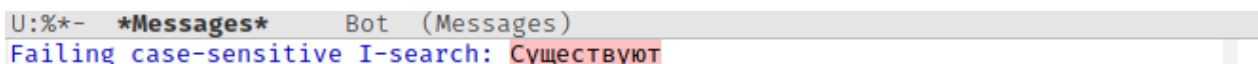


Figure 23: Режим поиска

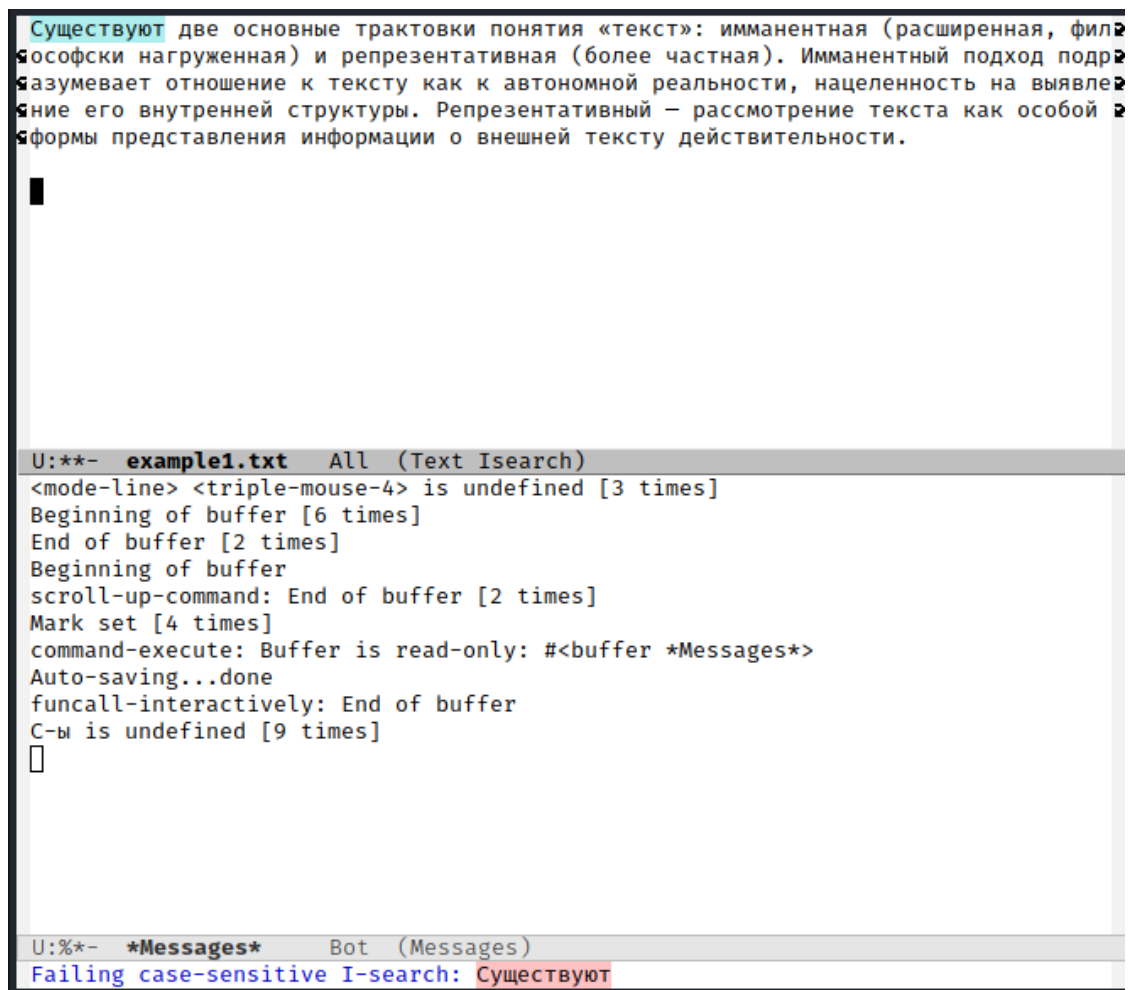


Figure 24: Результаты поиска

9.2. Переключимся между результатами поиска, нажимая «Ctrl-s». (рис. 25)

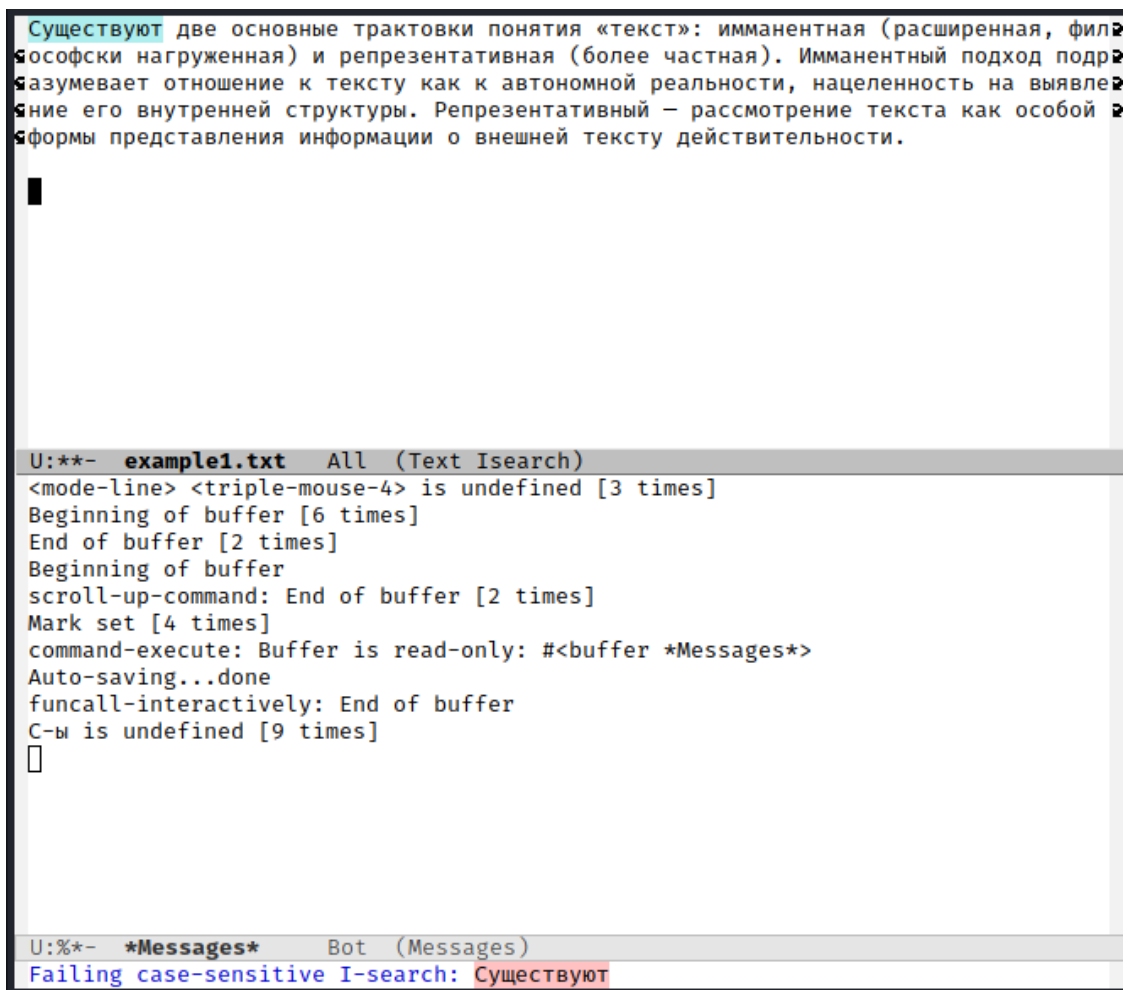


Figure 25: Переключение между результатами поиска

9.3. Выйдем из режима поиска, нажав «Ctrl-g».

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. 26) (рис. 27) (рис. 28) (рис. 29)

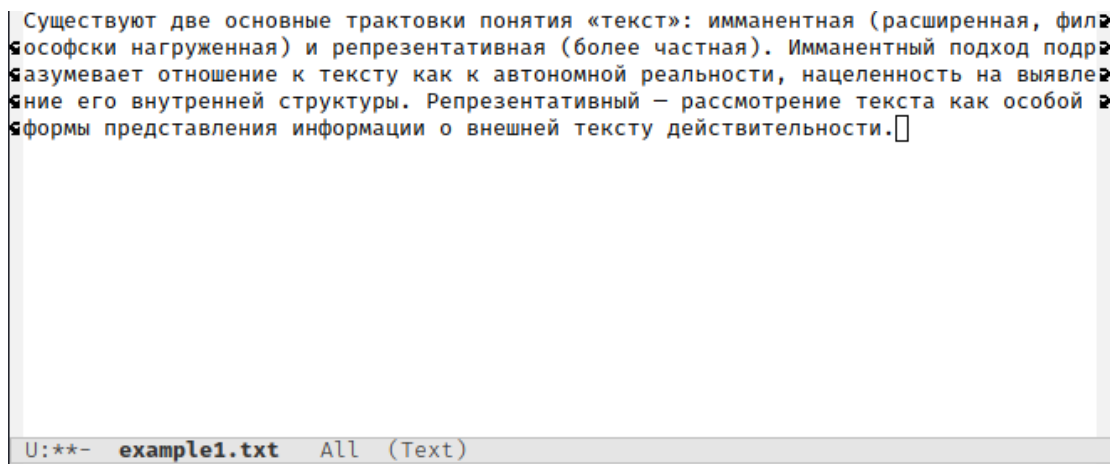


Figure 26: Ввод текста, который нужно заменить

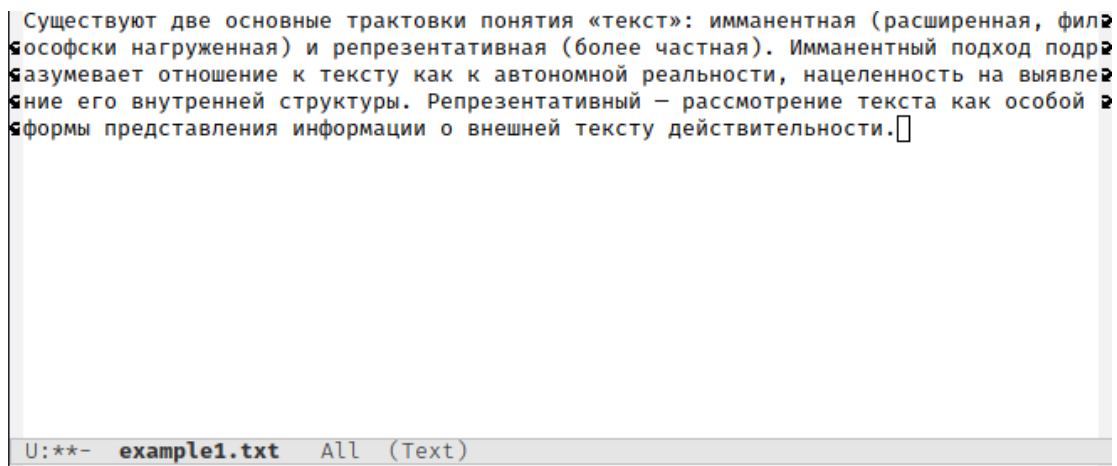


Figure 27: Ввод текста для замены

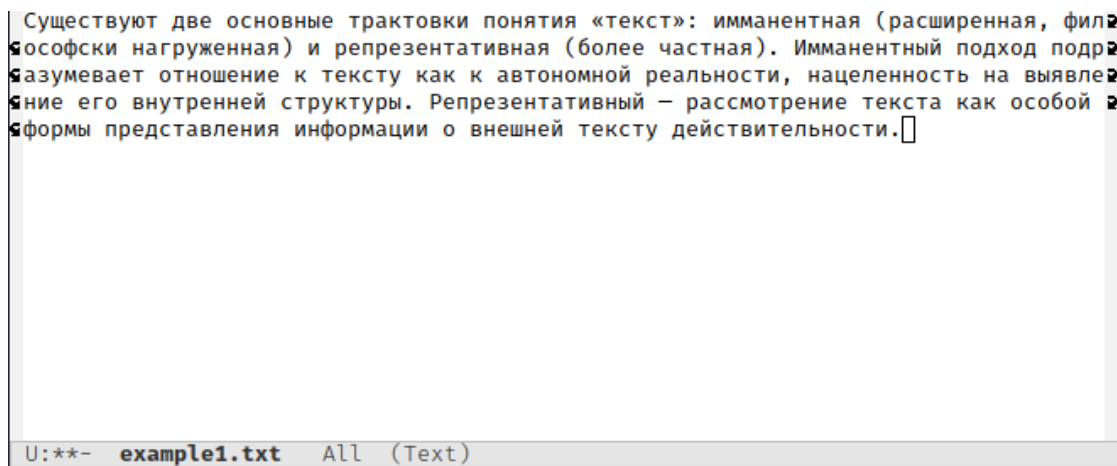
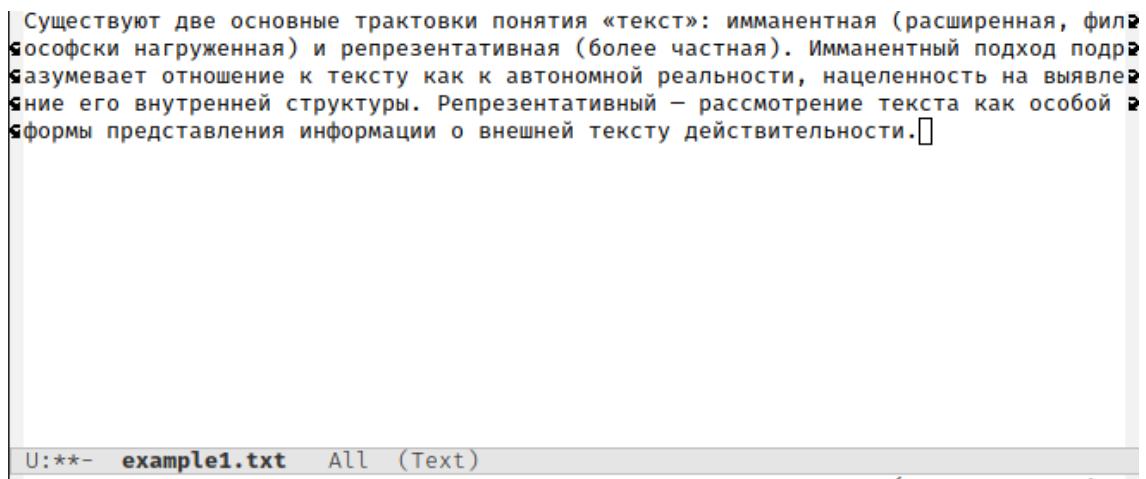


Figure 28: Подтверждение замены



Существуют две основные трактовки понятия «текст»: имманентная (расширенная, философски нагруженная) и репрезентативная (более частная). Имманентный подход подразумевает отношение к тексту как к автономной реальности, нацеленность на выявление его внутренней структуры. Репрезентативный – рассмотрение текста как особой формы представления информации о внешней тексту действительности.

U:**~ example1.txt All (Text)

Figure 29: Итоговая замена

9.5. Пробуем другой режим поиска, нажав «Alt-s о». Но у меня не сработала эта комбинация, поэтому я воспользовалась «Alt-x occur». (рис. 30)

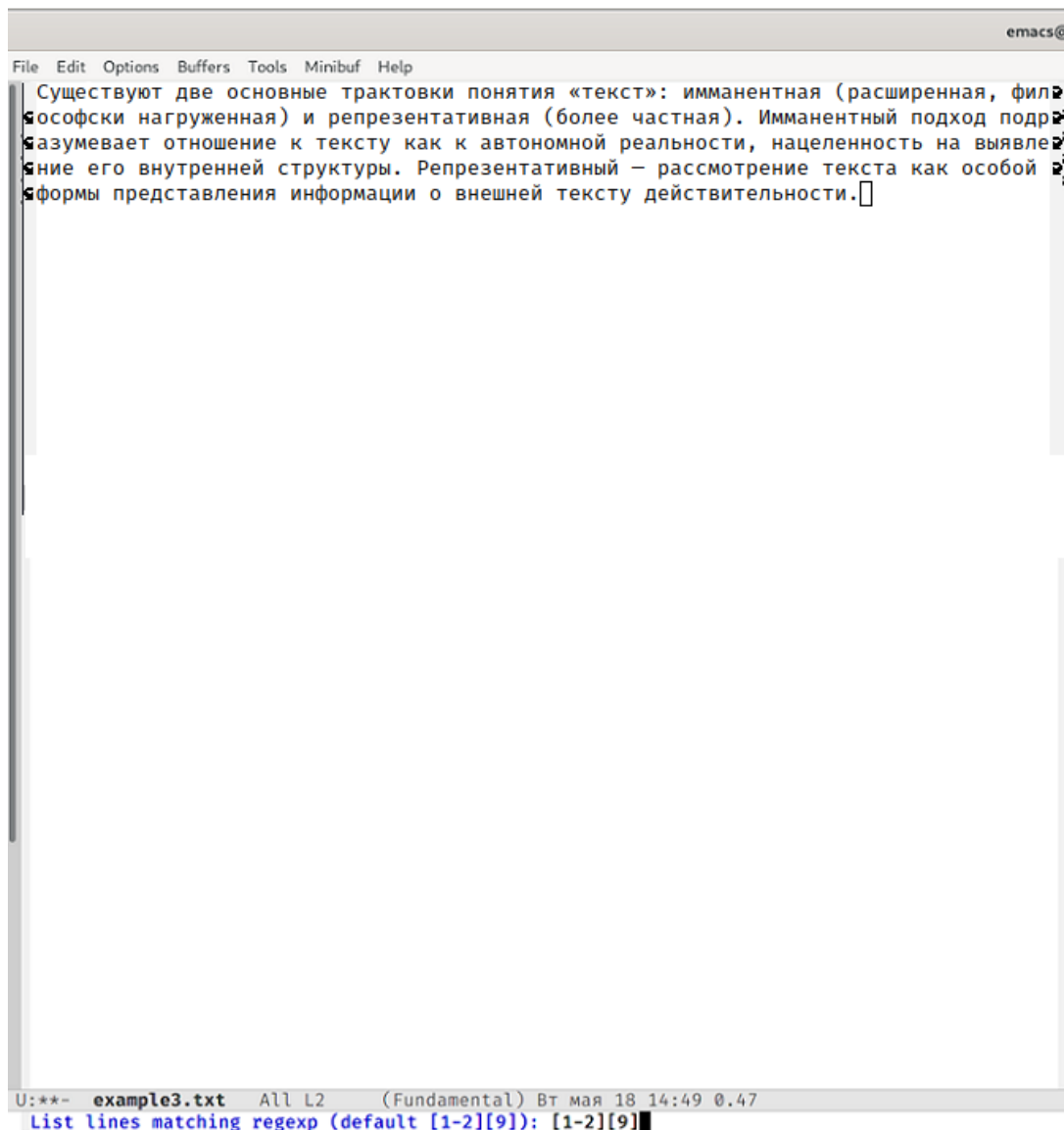


Figure 30: Другой режим поиска

Данный вид поиска отличается от обычного тем, что тут считывается строка поиска, которая трактуется как регулярное выражение, и не осуществляется поиск точного совпадения в тексте буфера. Регулярное выражение – это образец, который обозначает набор строк, возможно, и неограниченный набор.

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 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 открыться 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:
 - «scratch»(буфер для несохраненного текста)
 - «Messages»(журнал ошибок, включающий также информацию, которая появляется в области EchoArea)

- «GNU Emacs»(справочный буфер о редакторе)
- 6) C-с | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и нажимаю «|»

C-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 для меня является редактор emacs, так как в нем проще открывать другие файлы, можно использовать сразу несколько окон, нет «Командного режима», «Режима ввода», «Режима командной строки», которые являются в какой-то степени неудобными и непривычными.

Выводы

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

Библиография

1. https://esystem.rudn.ru/pluginfile.php/1142084/mod_resource/content/2/006-lab_vi.pdf
2. Кулябов Д.С. Операционные системы: лабораторные работы: учебное пособие / Д.С. Кулябов, М.Н. Геворкян, А.В. Королькова, А.В. Демидова. — М. : Изд-во РУДН, 2016. — 117 с. — ISBN 978-5-209-07626-1 : 139.13; То же [Электронный ресурс]. — URL: <http://lib.rudn.ru/MegaPro2/Download/MObject/6118>.
3. Робачевский А.М. Операционная система UNIX [текст] : Учебное пособие / А.М. Робачевский, С.А. Немнюгин, О.Л. Стесик. — 2-е изд., перераб. и доп. — СПб. : БХВ-Петербург, 2005, 2010. — 656 с. : ил. — ISBN 5-94157-538-6 : 164.56. (ЕТ 60)
4. Таненбаум Эндрю. Современные операционные системы [Текст] / Э. Таненбаум. — 2-е изд. — СПб. : Питер, 2006. — 1038 с. : ил. — (Классика Computer Science). — ISBN 5-318-00299-4 : 446.05. (ЕТ 50)