Отчёт по лабораторной работе 9

Операционные системы

Дьяконова Софья Александровна

Содержание

# 1 Цель работы

Цель работы – познакомиться с операционной системой Linux, получить практические навыки работы с редактором Emacs.

# 2 Задание

1. Открыть emacs.
2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).
3. Наберите текст: 1 #!/bin/bash 2 HELL=Hello 3 function hello { 4 LOCAL HELLO=World 5 echo $HELLO 6 } 7 echo $HELLO 8 hello
4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).
5. Проделать с текстом стандартные процедуры редактирования, каждое действие долж- но осуществляться комбинацией клавиш. 5.1. Вырезать одной командой целую строку (С-k). 5.2. Вставить эту строку в конец файла (C-y). 5.3. Выделить область текста (C-space). 5.4. Скопировать область в буфер обмена (M-w). 5.5. Вставить область в конец файла. 5.6. Вновь выделить эту область и на этот раз вырезать её (C-w). 5.7. Отмените последнее действие (C-/).
6. Научитесь использовать команды по перемещению курсора. 6.1. Переместите курсор в начало строки (C-a). 6.2. Переместите курсор в конец строки (C-e). 6.3. Переместите курсор в начало буфера (M-<). 6.4. Переместите курсор в конец буфера (M->).
7. Управление буферами. 7.1. Вывести список активных буферов на экран (C-x C-b). 7.2. Переместитесь во вновь открытое окно (C-x) o со списком открытых буферов и переключитесь на другой буфер. 7.3. Закройте это окно (C-x 0). 7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b).
8. Управление окнами. 8.1. Поделите фрейм на 4 части: разделите фрейм на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по горизонтали (C-x 2). 8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста.
9. Режим поиска 9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте. 9.2. Переключайтесь между результатами поиска, нажимая C-s. 9.3. Выйдите из режима поиска, нажав C-g. 9.4. Перейдите в режим поиска и замены (M-%), введите текст, который следует найти и заменить, нажмите Enter , затем введите текст для замены. После того как будут подсвечены результаты поиска, нажмите ! для подтверждения замены. 9.5. Испробуйте другой режим поиска, нажав M-s o. Объясните, чем он отличается от обычного режима?

# 3 Теоретическое введение

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; 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. Первая версия редактора Emacs была написана в 70-х годах 20-го столетия Richard Stallman (Ричардом Столманом) как набор макросов для редактора TECO . В дальнейшем, уже будучи основателем Фонда Свободного программного обеспечения Free Software Foundation и проекта GNU, Stallman разработал GNU Emacs в развитие оригинального Emacs и до сих пор сопровождает эту программу. Emacs является одним из старейших редакторов. Он использовался тысячами программистов на протяжении последних 20 с лишним лет, для него создано много дополнительных пакетов расширений. Эти дополнения позволяют делать с помощью Emacs такие вещи, которые Stallman , вероятно, даже не считал возможными в начале своей работы над редактором.

# 4 Выполнение лабораторной работы

1. Открываю emacs (рис.1. [1](#fig:001)).

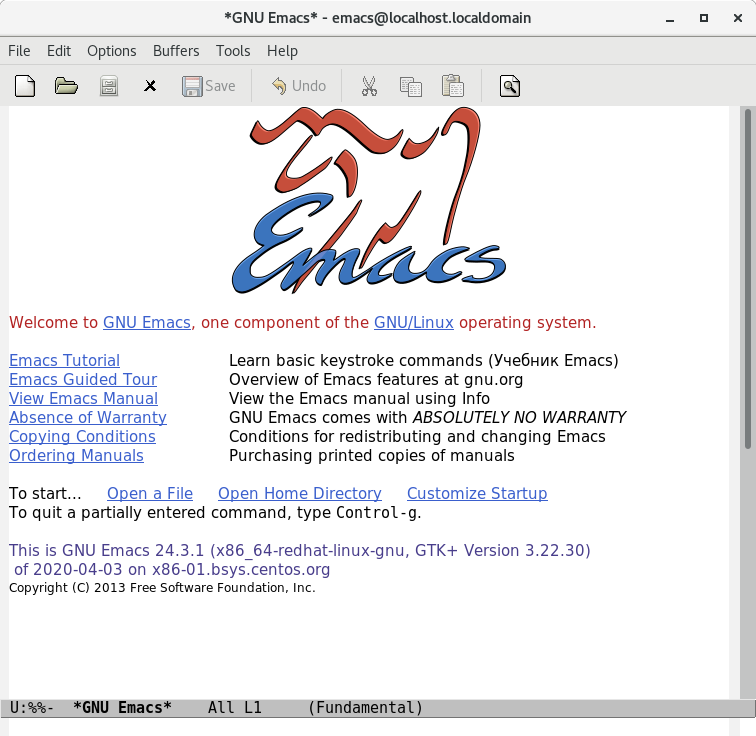


Figure 1: emacs

1. Создаю файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f) (рис.1. [1](#fig:001)).

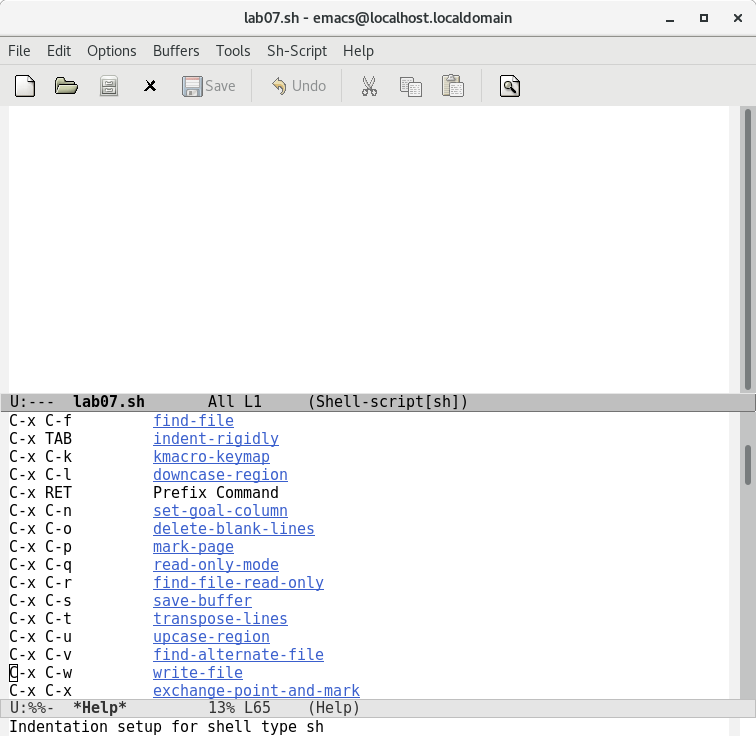


Figure 2: lab07.sh

1. Набераю текст: 1 #!/bin/bash 2 HELL=Hello 3 function hello { 4 LOCAL HELLO=World 5 echo $HELLO 6 } 7 echo $HELLO 8 hello (рис.3. [3](#fig:003))

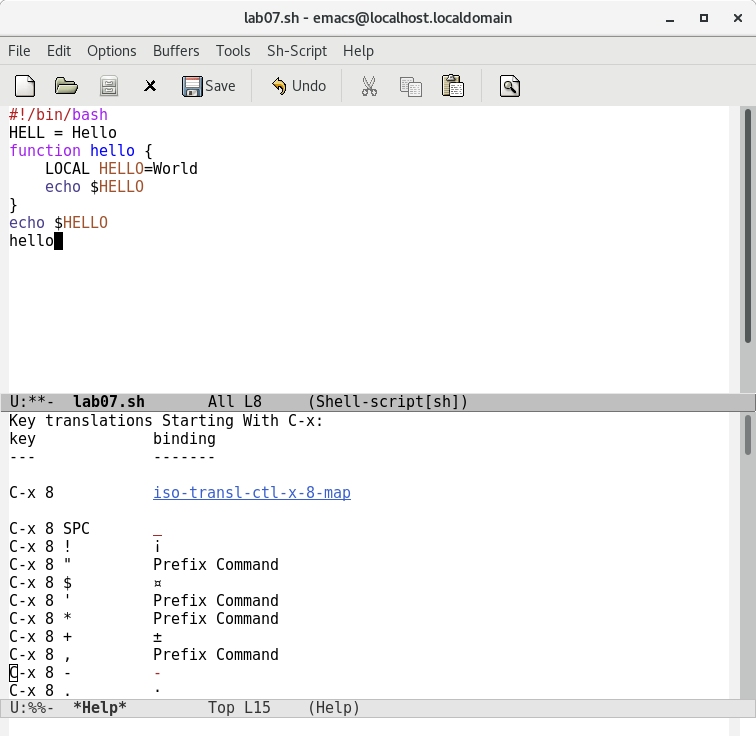


Figure 3: hello

1. Сохраняю файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s) .
2. Проделываю с текстом стандартные процедуры редактирования, каждое действие осуществляется комбинацией клавиш. 5.1. Вырезаю одной командой целую строку (С-k) (рис.4. [4](#fig:004)).

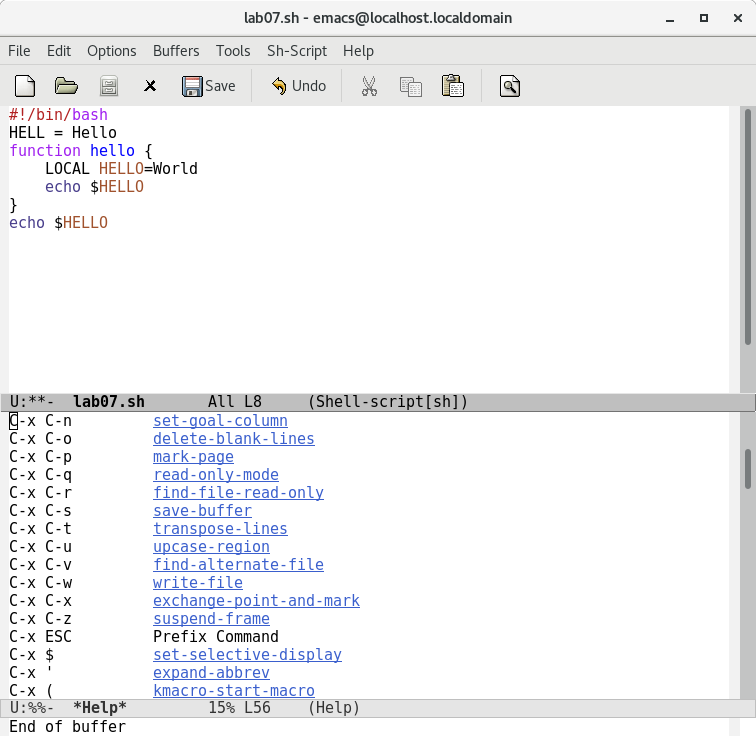


Figure 4: С-k

5.2. Вставляю эту строку в конец файла (C-y) (рис.5. @fig:005).

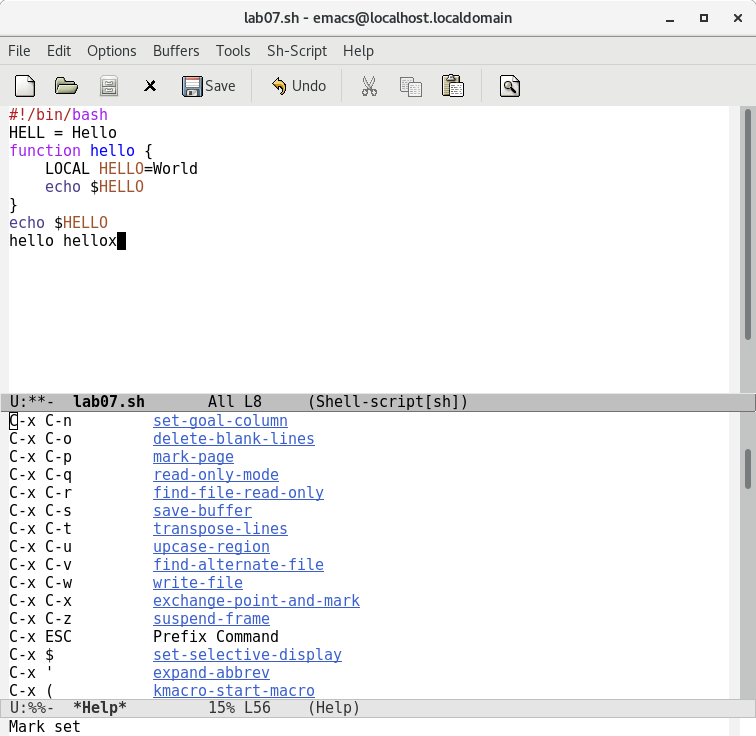


Figure 5: С-y

5.3. Выделяю область текста (C-space).  
5.4. Копирую область в буфер обмена (M-w).  
5.5. Вставляю область в конец файла (рис.6. @fig:006).

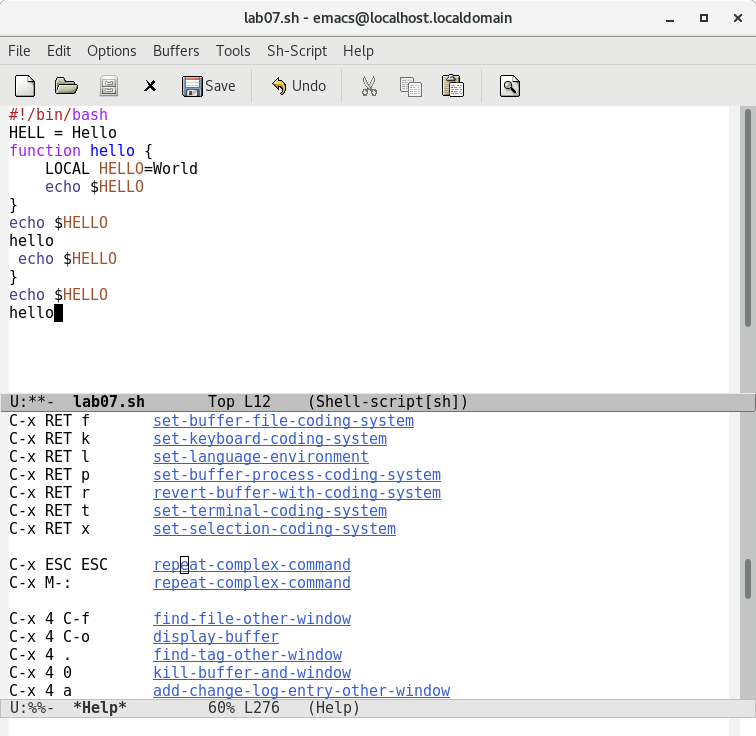
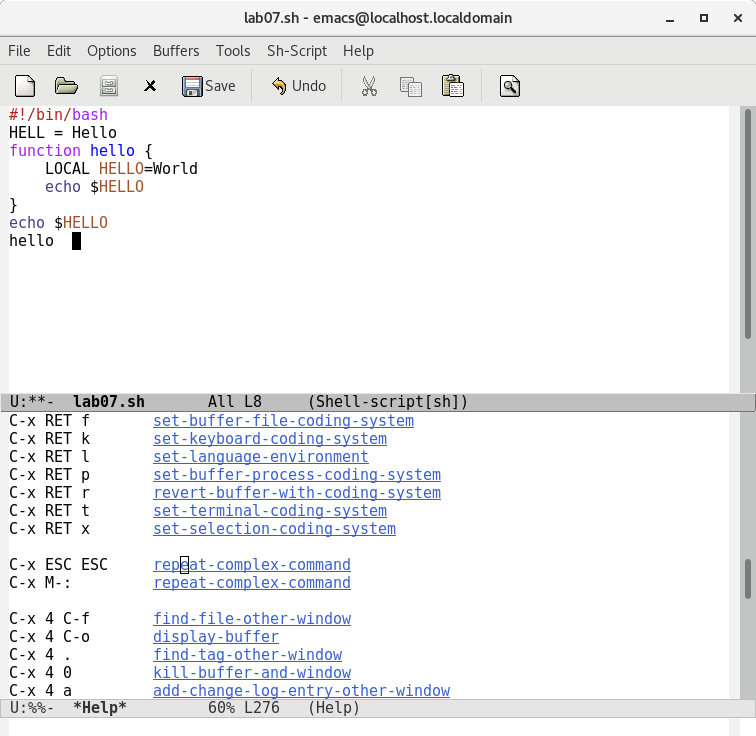


Figure 6: Область в конце файла

5.6. Вновь выделяю эту область и на этот раз вырезаю её (C-w) (рис.7. @fig:007).

 5.7. Отменяю последнее действие (C-/) (рис.8. [7](#fig:008)).

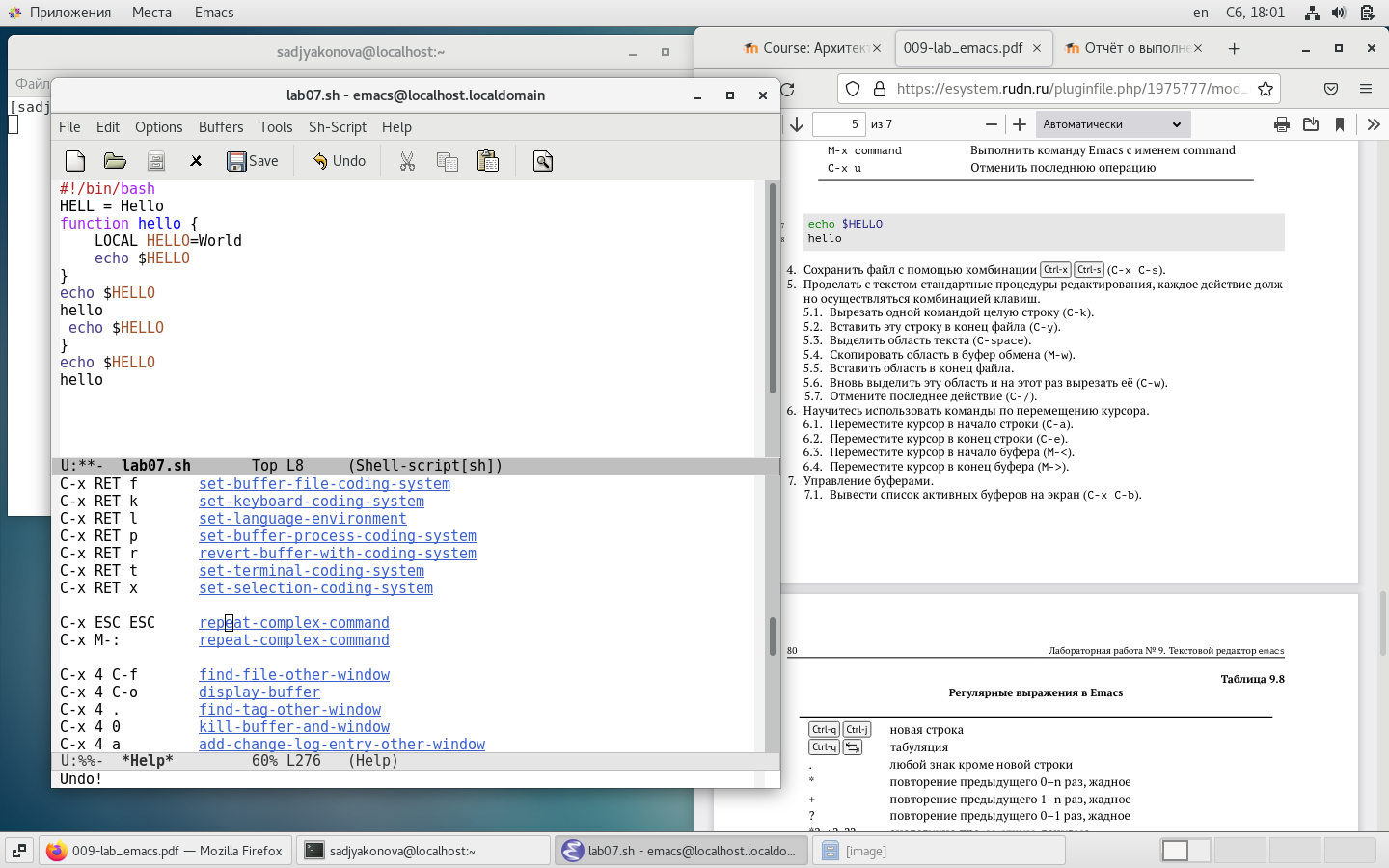


Figure 7: С-/

1. Учусь использовать команды по перемещению курсора. 6.1. Перемещаю курсор в начало строки (C-a). 6.2. Перемещаю курсор в конец строки (C-e). 6.3. Перемещаю курсор в начало буфера (M-<). 6.4. Перемещаю курсор в конец буфера (M->).
2. Управление буферами. 7.1. Вывожу список активных буферов на экран (C-x C-b) (рис.9. [8](#fig:009)).

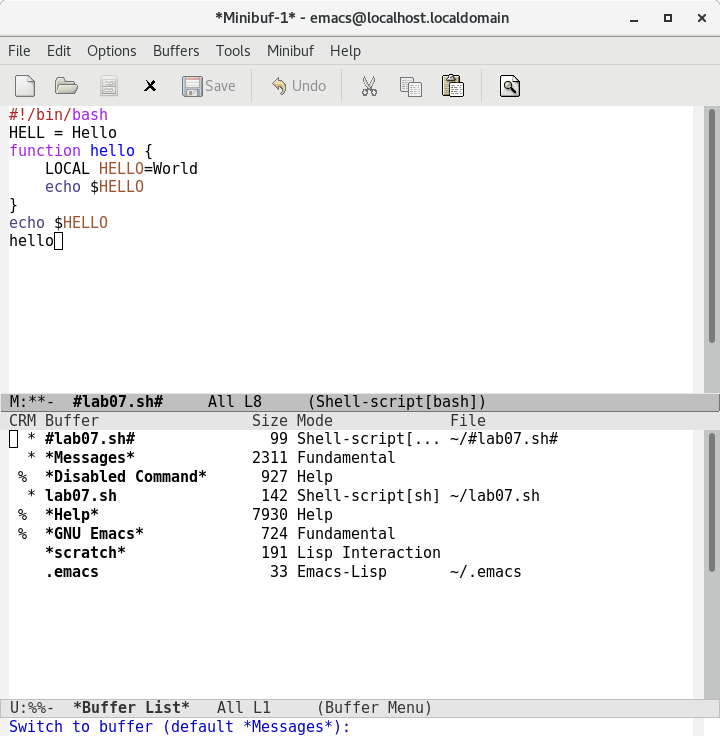


Figure 8: С-x С-b

7.2. Перемещаюсь во вновь открытое окно (C-x) o со списком открытых буферов и переключаюсь на другой буфер.  
7.3. Закрываю это окно (C-x 0).  
7.4. Теперь вновь переключаюсь между буферами, но уже без вывода их списка на экран (C-x b).

1. Управление окнами. 8.1. Делю фрейм на 4 части (рис.10. [9](#fig:010)).



Figure 9: 4 фрейма

8.2. В каждом из четырёх созданных окон открываю новый буфер (файл) и ввожу несколько строк текста (рис.11. @fig:011).

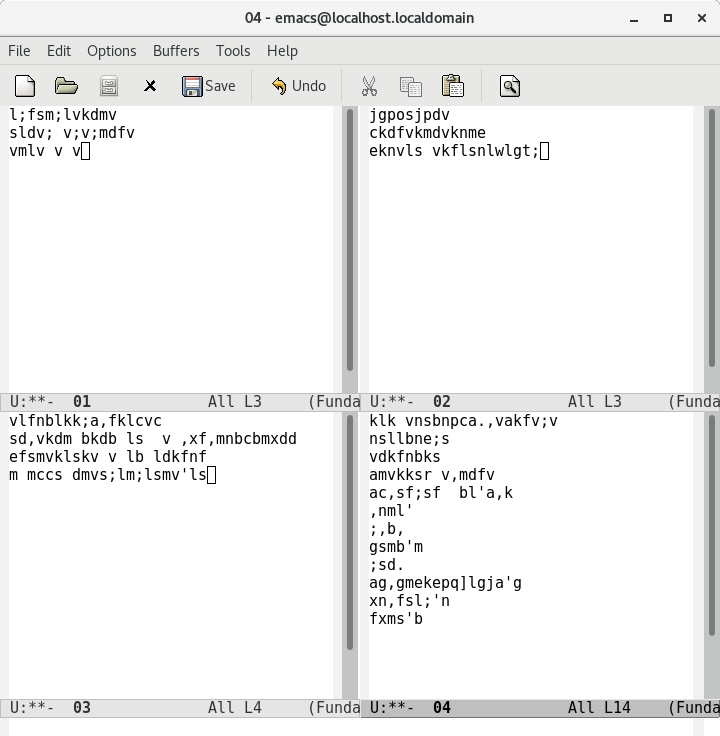


Figure 10: Какой-то текст

1. Режим поиска 9.1. Переключаюсь в режим поиска (C-s) и нахожу несколько слов, присутствующих в тексте (рис.12. [11](#fig:012)).

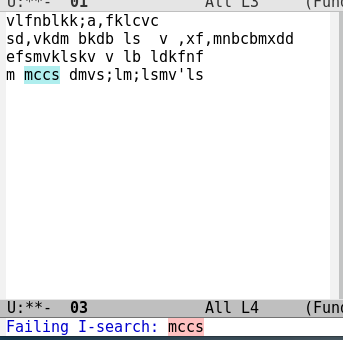


Figure 11: С-s

9.2. Переключаюсь между результатами поиска, нажимая C-s.  
9.3. Выхожу из режима поиска, нажав C-g.  
9.4. Перехожу в режим поиска и замены (M-%), ввожу текст, который следует найти и заменить, нажимаю Enter , затем ввожу текст для замены. После того как будут подсвечены результаты поиска, нажимаю ! для подтверждения замены.   
9.5. Пробую другой режим поиска, нажав M-s o. В дополнение к обычному режиму появляется вывод строк с найденными элэментами и необходимо нажать Enter.

#Контрольные вопросы 1. Кратко охарактеризуйте редактор emacs. Emacs — один из наиболее мощных и широко распространённых редакторов, используемых в мире UNIX. Написан на языке высокого уровня Lisp. 2. Какие особенности данного редактора могут сделать его сложным для освоения новичком? Большое разнообразие сложных комбинаций клавиш, которые необходимы для редактирования файла и в принципе для работа с Emacs. 3. Своими словами опишите, что такое буфер и окно в терминологии emacs’а. Буфер - это объект в виде текста. Окно - это прямоугольная область, в которой отображен буфер. 4. Можно ли открыть больше 10 буферов в одном окне? Да, можно. 5. Какие буферы создаются по умолчанию при запуске emacs? Emacs использует буферы с именами, начинающимися с пробела, для внутренних целей. Отчасти он обращается с буферами с такими именами особенным образом — например, по умолчанию в них не записывается информация для отмены изменений. 6. Какие клавиши вы нажмёте, чтобы ввести следующую комбинацию C-c | и C-c C-|? Ctrl + c, а потом | и Ctrl + c Ctrl + | 7. Как поделить текущее окно на две части? С помощью команды Ctrl + x 3 (по вертикали) и Ctrl + x 2 (по горизонтали). 8. В каком файле хранятся настройки редактора emacs? Настройки emacs хранятся в файле . emacs, который хранится в домашней дирректории пользователя. Кроме этого файла есть ещё папка . emacs. 9. Какую функцию выполняет клавиша и можно ли её переназначить? Выполняет функцию стереть, думаю можно переназначить. 10. Какой редактор вам показался удобнее в работе vi или emacs? Поясните почему. Для меня удобнее был редактор Emacs, так как у него есть командая оболочка. А vi открывается в терминале, и выглядит своеобразно.

# 5 Выводы

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