Отчёт по лабораторной работе №9

Текстовой редактор emacs

Щербак Маргарита Романовна

2022

# 1 Цель работы:

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

# 2 Теоретическое введение:

Emacs представляет собой мощный экранный редактор текста, написанный на языке высокого уровня Elisp.  
Буфер — объект, представляющий какой-либо текст. Буфер может содержать что угодно, например, результаты компиляции программы или встроенные подсказки. Практически всё взаимодействие с пользователем, в том числе интерактивное, происходит посредством буферов.  
Фрейм соответствует окну в обычном понимании этого слова. Каждый фрейм содержит область вывода и одно или несколько окон Emacs.  
Окно — прямоугольная область фрейма, отображающая один из буферов. Каждое окно имеет свою строку состояния, в которой выводится следующая информация: название буфера, его основной режим, изменялся ли текст буфера и как далеко вниз по буферу расположен курсор. Каждый буфер находится только в одном из возможных основных режимов. Существующие основные режимы включают режим Fundamental (наименее специализированный), режим Text, режим Lisp, режим С, режим Texinfo и другие. Под второстепенными режимами понимается список режимов, которые включены в данный момент в буфере выбранного окна.  
Область вывода — одна или несколько строк внизу фрейма, в которой Emacs выводит различные сообщения, а также запрашивает подтверждения и дополнительную информацию от пользователя.  
Минибуфер используется для ввода дополнительной информации и всегда отображается в области вывода.  
Точка вставки — место вставки (удаления) данных в буфере.

# 3 Выполнение лабораторной работы:

1. Ознакомились с теоретическим материалом, редактором emacs. Открыли emacs. (Рис. 1).

Рис. 1: Открываем текстовой редактор emacs

Рис. 1: Открываем текстовой редактор emacs

1. Создали файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f). (Рис. 2).

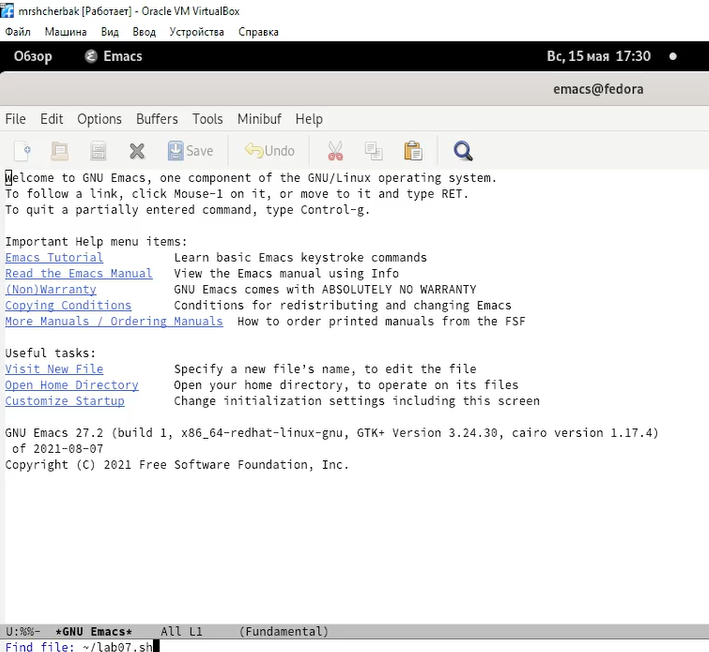


Рис. 2: Создание файла

1. Набрали текст (Рис. 3).

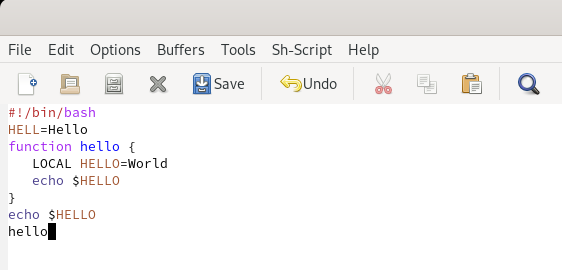


Рис. 3: Выполнение

1. Сохранили файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).
2. Проделали с текстом стандартные процедуры редактирования.

* вырезали одной командой целую строку “function hello {” (С-k). (Рис. 4).

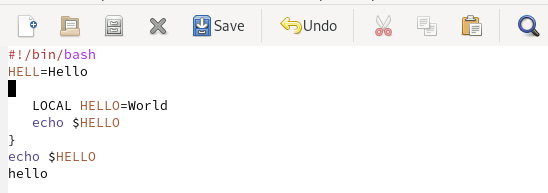


Рис. 4: Вырезали строку

* вставили эту строку в конец файла (C-y). (Рис. 5).

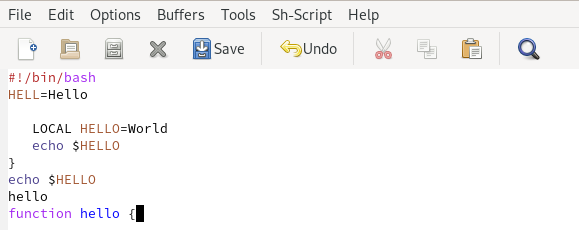


Рис. 5: Вставили вырезанную строку в конец

* выделили область текста (C-space).
* скопировали область в буфер обмена (M-w).
* вставили область в конец файла. (Рис. 6).

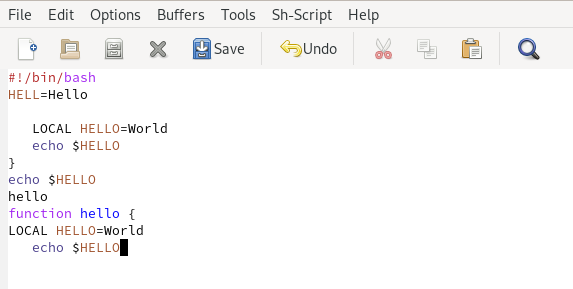


Рис. 6: Вставили вырезанную строку в конец

* выделили эту область и вырезали её (C-w). (Рис. 7).

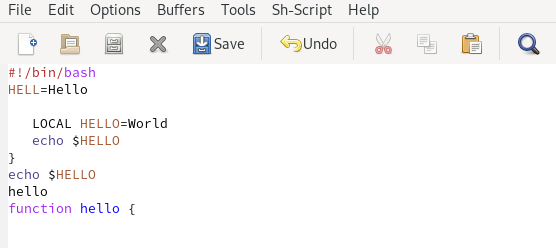


Рис. 7: Вырезали выделенную область

* отменили последнее действие (C-/). (Рис. 8).

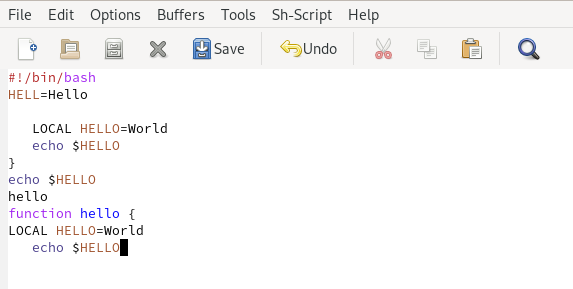


Рис. 8: Вернули вырезанную область

1. Научились использовать команды по перемещению курсора. (Рис. 9).

* переместили курсор в начало строки (C-a).
* переместили курсор в конец строки (C-e).
* переместили курсор в начало буфера (M-<).
* переместили курсор в конец буфера (M->).

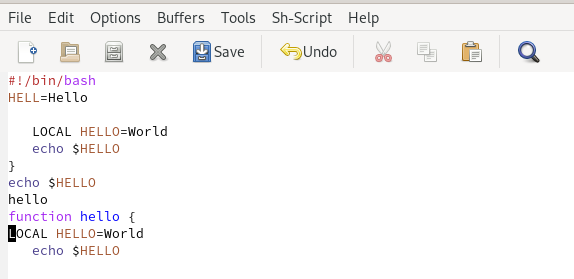


Рис. 9: Выполнение

1. Управление буферами.

* вывели список активных буферов на экран (C-x C-b) (Рис. 10).

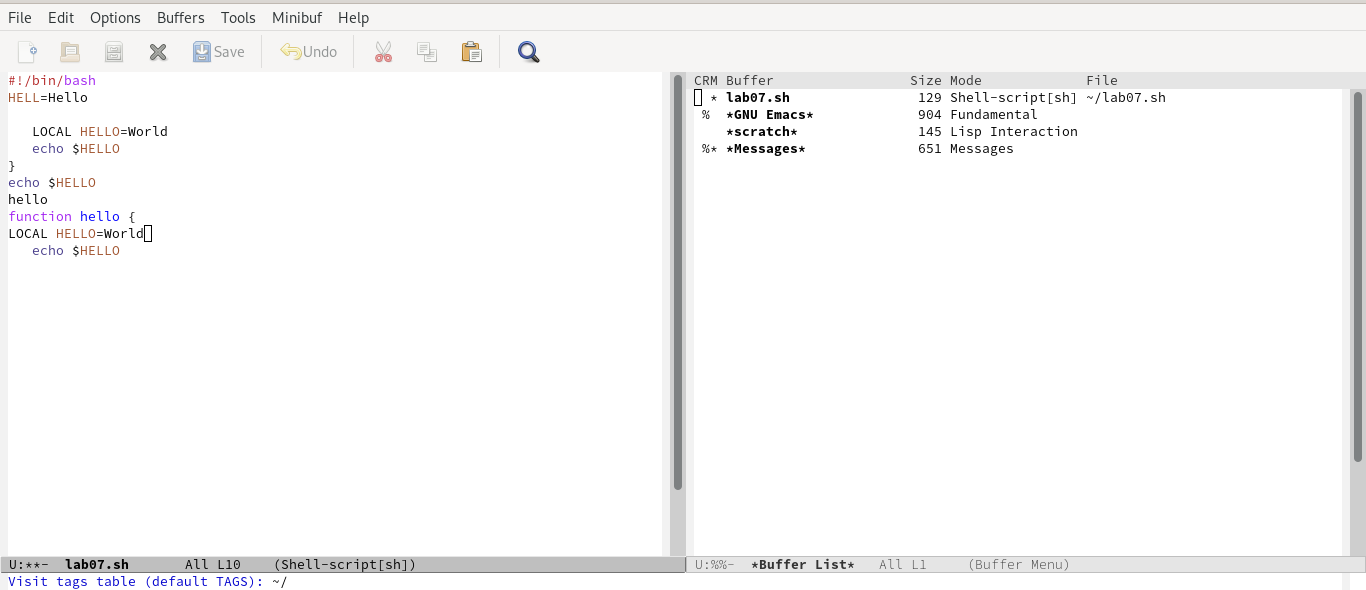


Рис. 10: Список буферов

* переместились во вновь открытое окно (C-x o) со списком открытых буферов и переключились на другой буфер. (Рис. 11).

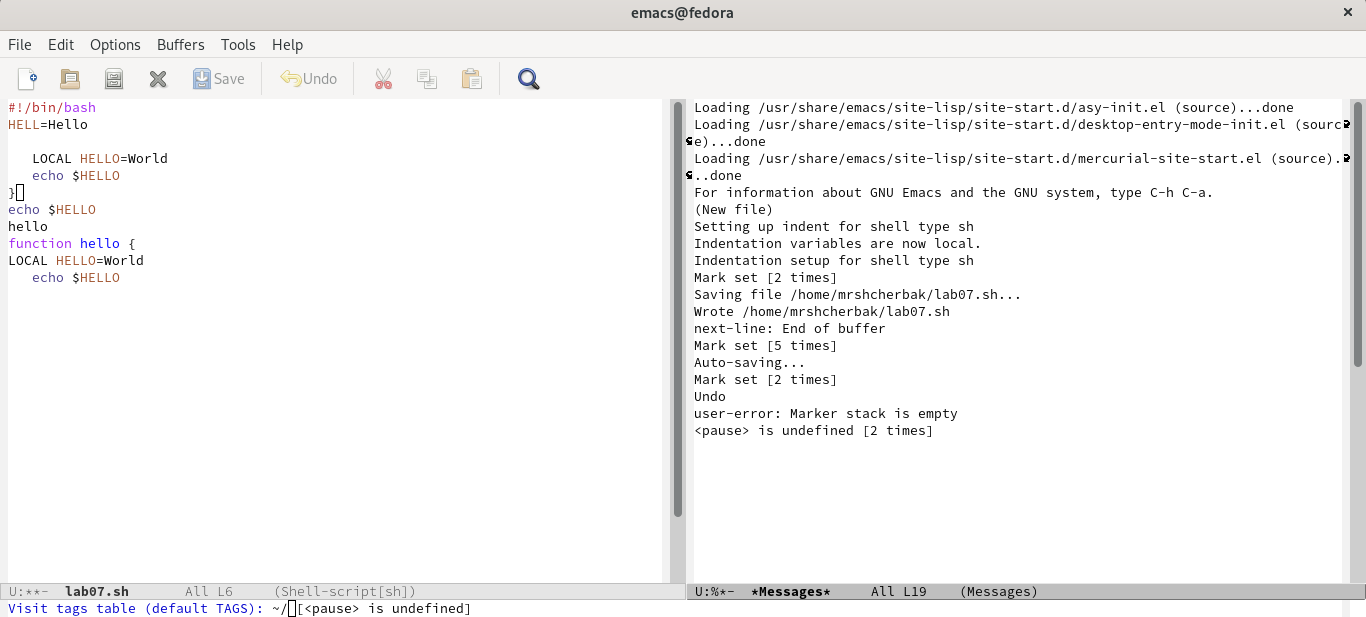


Рис. 11: Переключились на другой буфер

* закрыли это окно (C-x 0).
* переключались между буферами, но уже без вывода их списка на экран (C-x b). (Рис. 12).

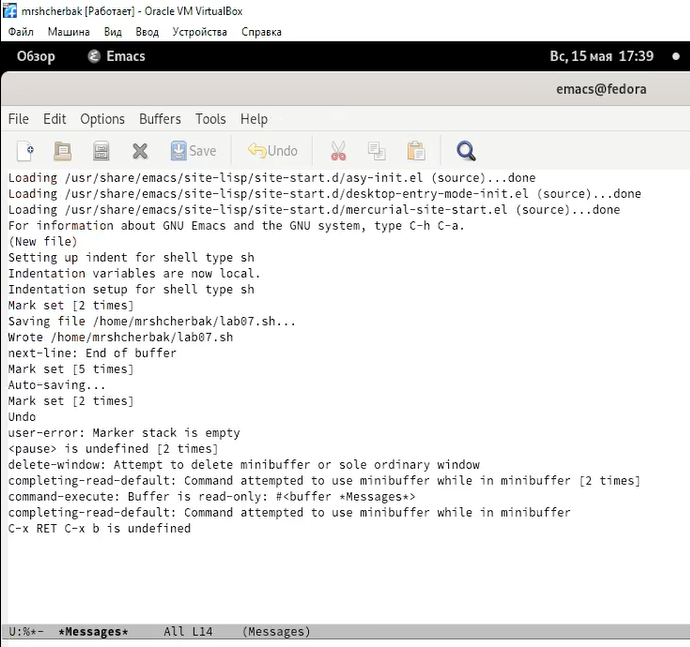


Рис. 12: Переключились на другой буфер без списка буферов

1. Управление окнами.

* поделили фрейм на 4 части: разделили фрейм на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по горизонтали (C-x 2) (Рис. 13 - Рис. 14).

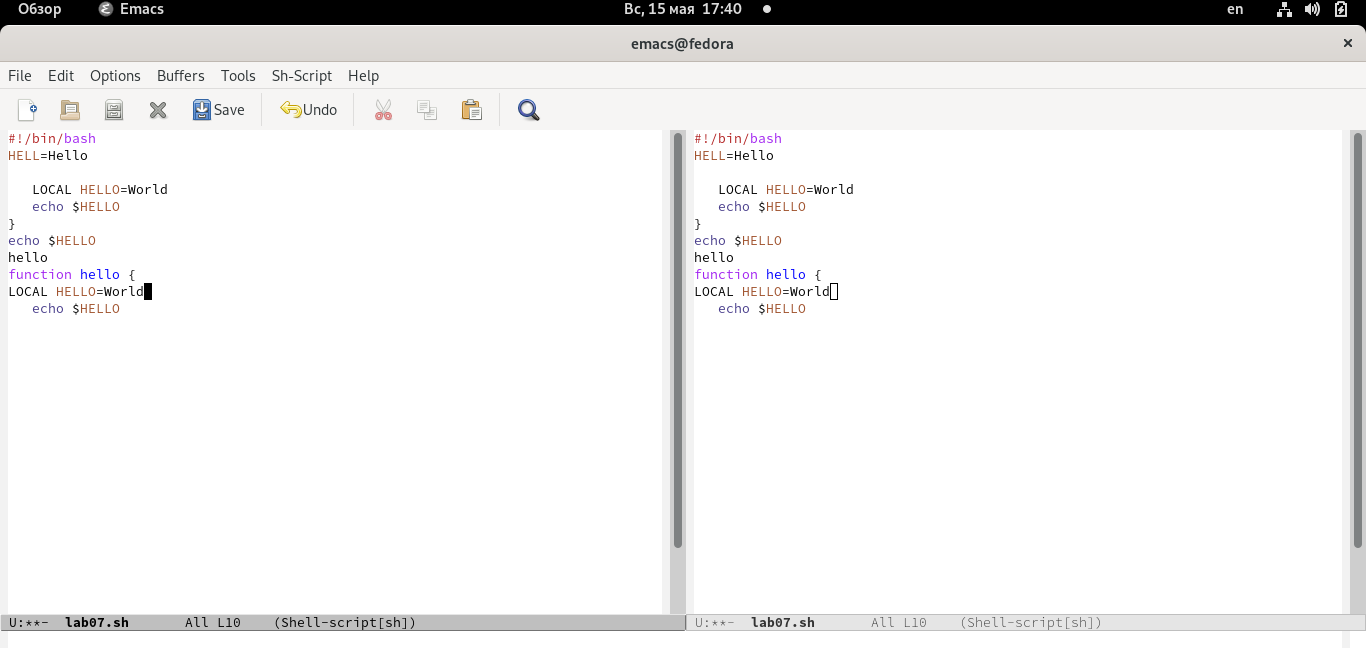


Рис. 13: разделили фрейм на два окна по вертикали

![Рис. 14: каждое из окон на две части по горизонтали разделили](data:application/octet-stream;base64,)

Рис. 14: каждое из окон на две части по горизонтали разделили

* в каждом из четырёх созданных окон открыли новый буфер (файл) и ввели несколько строк текста. (Рис. 15 - Рис. 16).

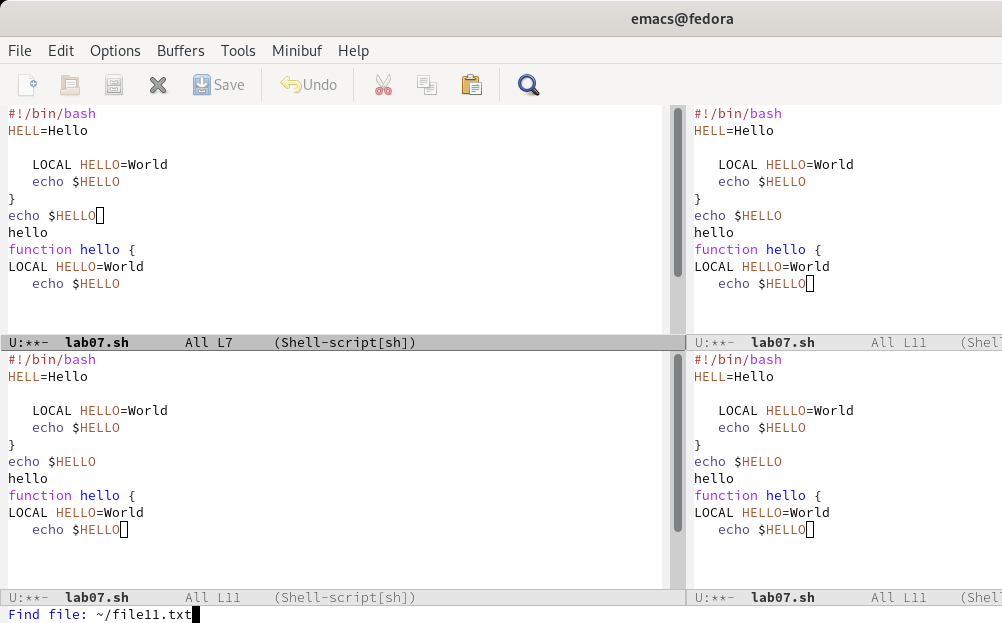


Рис. 15: Открываем файлы в окнах

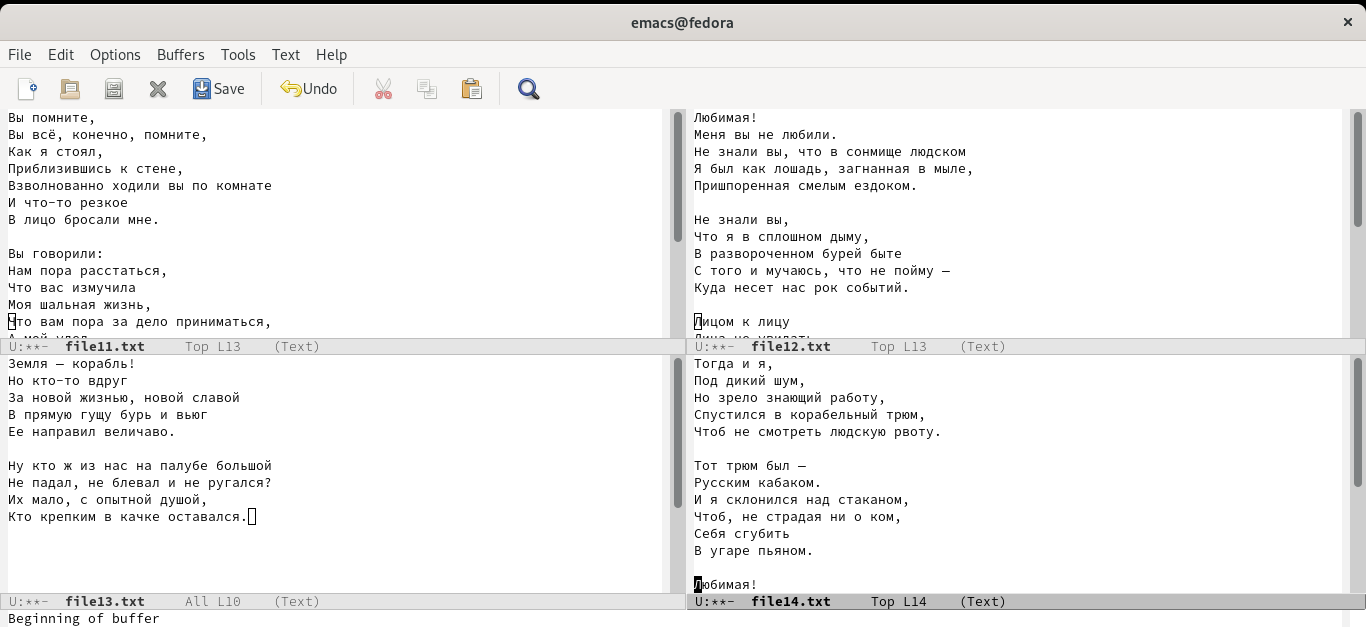


Рис. 16: Вводим текст в каждом файле

1. Режим поиска

* переключились в режим поиска (C-s) и нашли несколько слов, присутствующих в тексте. (Рис. 17).

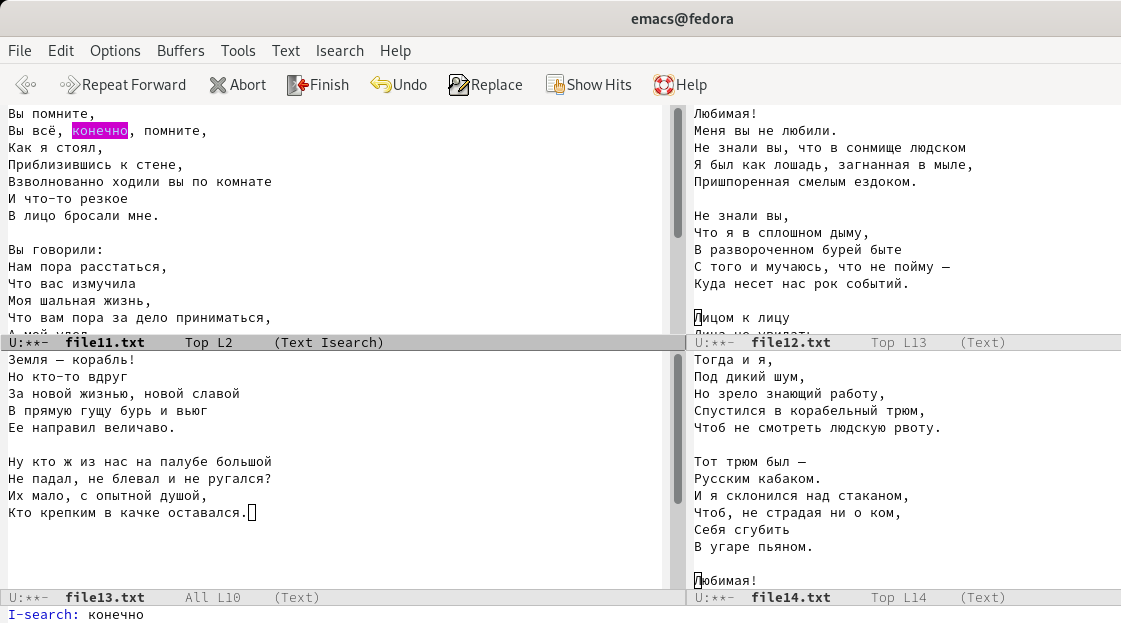


Рис. 17: Нашли слово “конечно”

* переключались между результатами поиска, нажимая C-s. (Рис. 18).

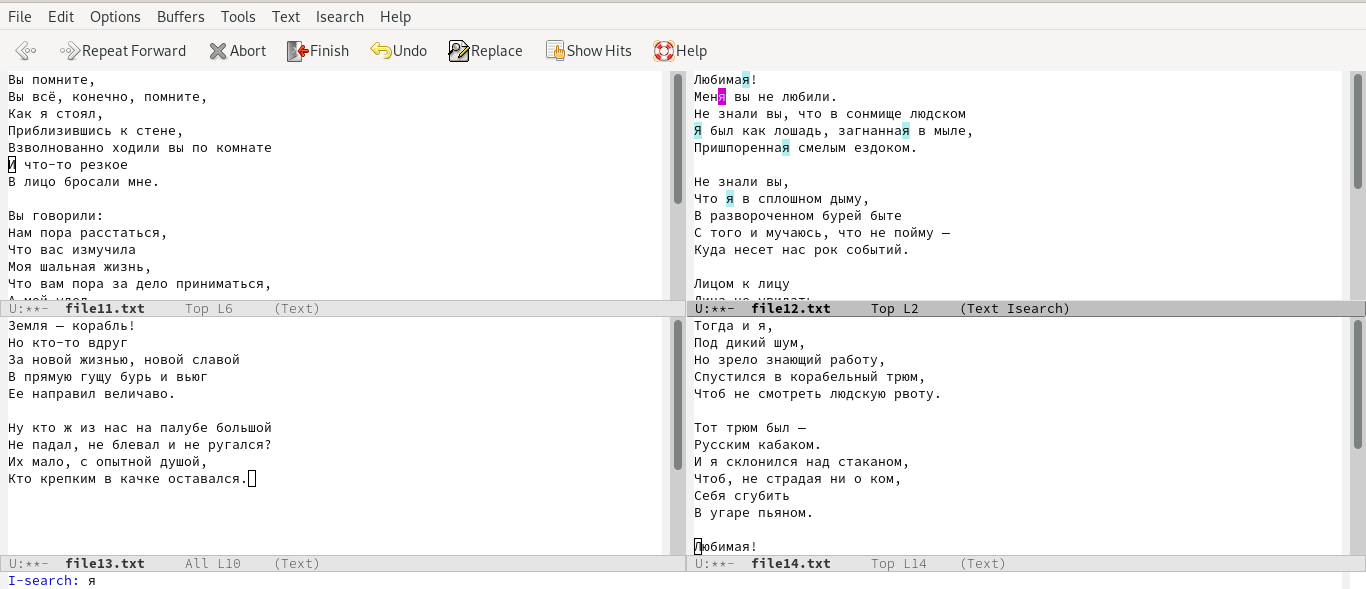


Рис. 18: Результаты поиска

* вышли из режима поиска, нажав C-g. (Рис. 19).

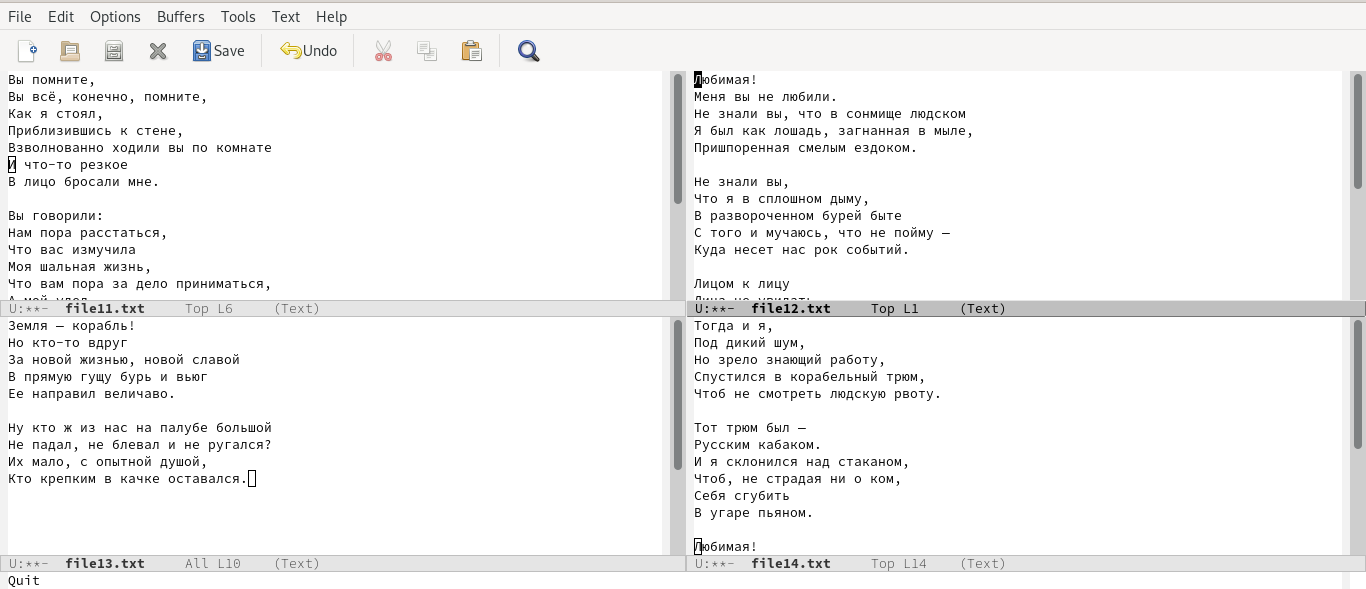


Рис. 19: Выход из режима поиска

* перешли в режим поиска и замены (Esc+Shift+%), ввели текст, который следует найти и заменить, нажали Enter , затем ввели текст для замены. Нажали ! для подтверждения замены. (Рис. 20 - Рис. 21).

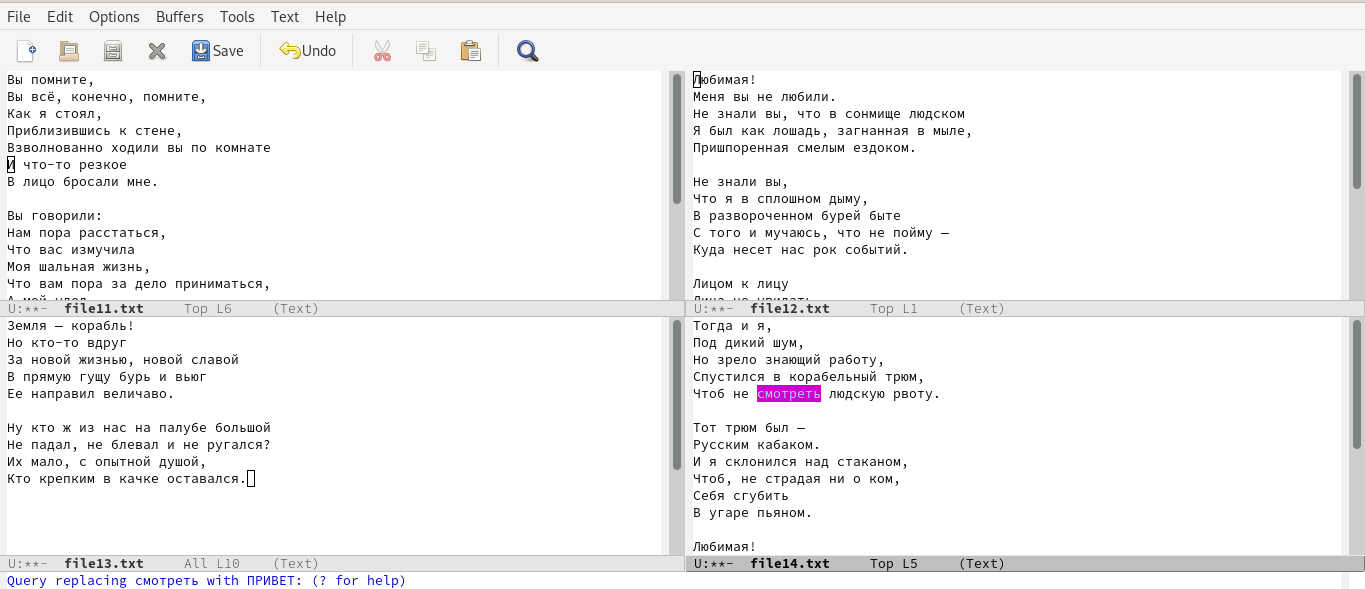


Рис. 20: Ввели слово, которое хотим найти и заменить

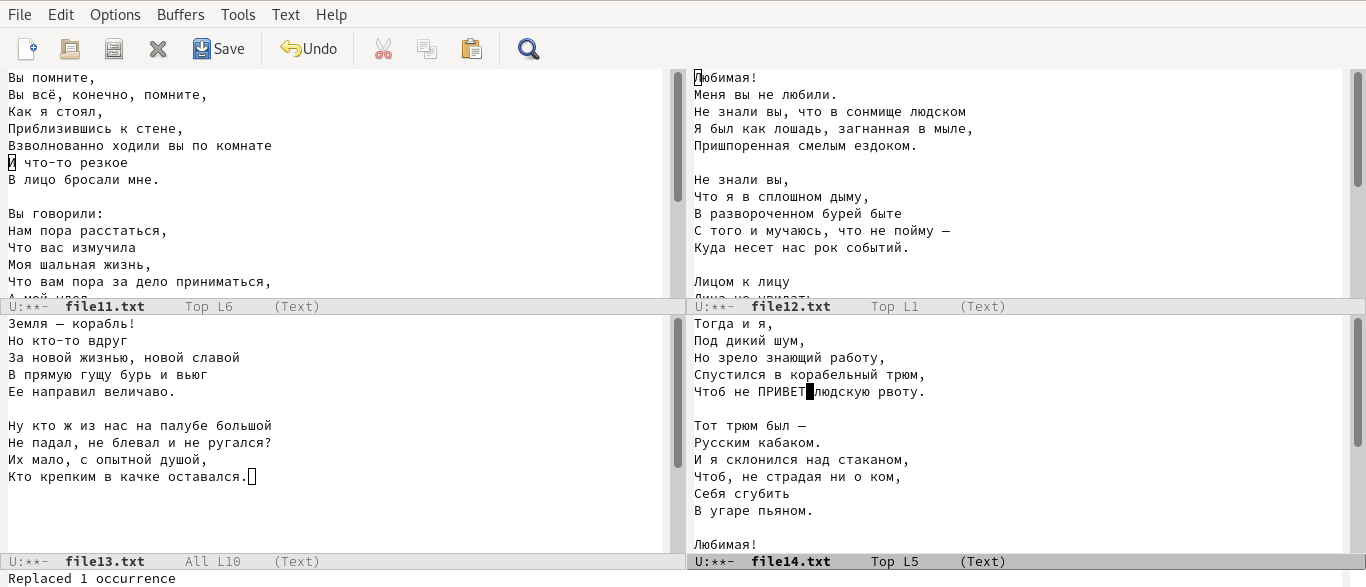


Рис. 21: Заменили слово “смотреть” на “ПРИВЕТ”

* испробовали другой режим поиска, нажав Alt-s o. (Рис. 22).

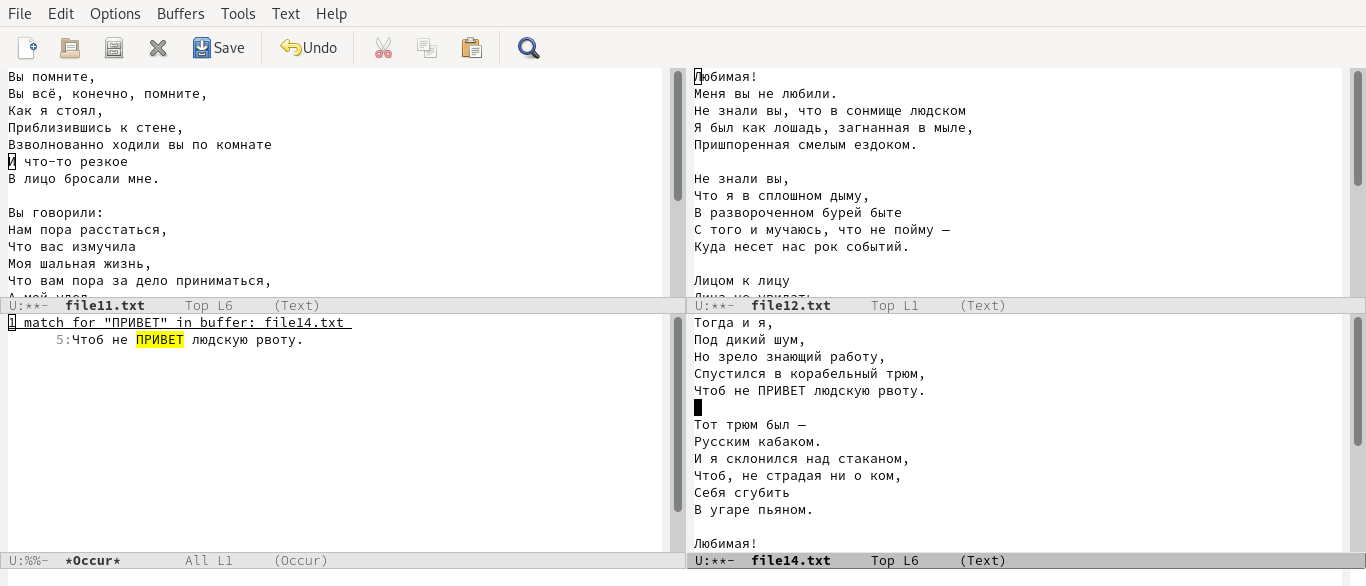


Рис. 22: Ищем слово “ПРИВЕТ” в file14.txt

Режим поиска и замены ALt-% показывает найденные совпадения искомого слова, а Alt-s o показывает название буфера (файла), номера строк, в которых содержится искомое слово, и совпадения. Оба выделяют цветом.

*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 освоенииданного 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 XWindow – разные вещи. Одно окно XWindow может быть разбито на несколько окон в смысле Emacs, в каждом из которых отображается отдельный буфер.
4. Не получится, так как окно - прямоугольная область фрейма, отображающая один из буферов, то есть 10 окон - 10 буферов и в одном окне 10 буферов нельзя открыть.
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) «GNUE macs»(справочный буфер о редакторе).
6. *C-c |* сначала, удерживая «ctrl»,нажимаю «c»,после отпускаю обе клавишии нажимаю «|»  
   *C-c C-|* 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 данная клавиша удаляет символ перед курсором, но в редакторе её можно переназначить. Для этого необхдимо 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. Emacs имеет значительно больше расширений, позволяющих выполнять задачи, которые связаны только с текстовым редактором, например, просмотр файловой системы или управление версиями, а также расширения, которые никоим образом не связаны с текстовым редактором, например, чтение RSS-каналов. Если нужна среда, а не просто текстовый редактор, Emacs будет лучше, чем Vim. Я также думаю, что Emacs гораздо легче освоить.

# 4 Выводы

Таким образом, в ходе ЛР№9 я познакомилась с операционной системой Linux. Получила практические навыки работы с редактором Emacs.