

Отчёт по лабораторной работе №9

Текстовый редактор emacs

Аскеров Александр Эдуардович

Содержание

1	Цель работы	4
2	Теоретическое введение	5
3	Выполнение лабораторной работы	6
4	Контрольные вопросы	14
5	Выводы	17

Список иллюстраций

3.1	Редактор emacs	6
3.2	Создание файла	7
3.3	Требуемый текст	7
3.4	Сохранение файла	7
3.5	Текст после изменений	8
3.6	Список активных буферов	9
3.7	Переключение на другой буфер	9
3.8	Переключение между буферами	10
3.9	Деление фрейма на четыре части	10
3.10	Текст в буфере в четырёх окнах	11
3.11	Поиск слов в тексте	12
3.12	Переключение между результатами поиска	12
3.13	Результат команды M-s o	13

1 Цель работы

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

2 Теоретическое введение

Буфер — объект, представляющий какой-либо текст.

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.

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. Откроем emacs.

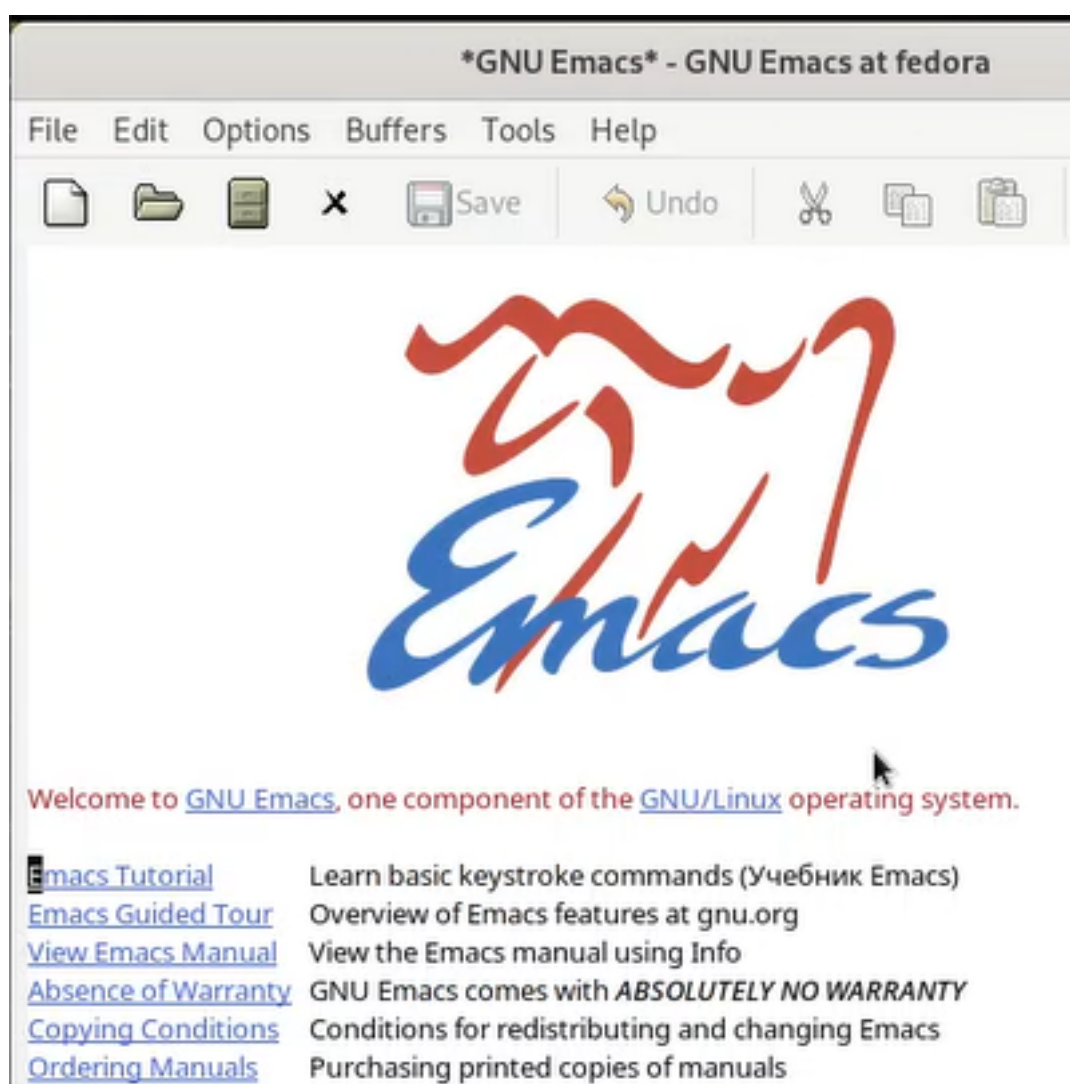


Рис. 3.1: Редактор emacs

2. Создадим файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f.

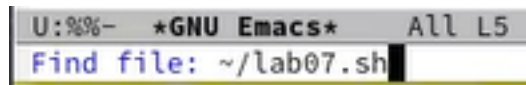


Рис. 3.2: Создание файла

3. Напечатаем следующий текст.

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

Рис. 3.3: Требуемый текст

4. Сохраним файл с помощью комбинации Ctrl-x Ctrl-s.

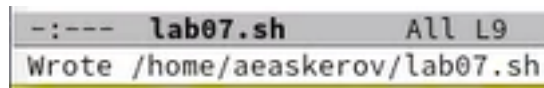


Рис. 3.4: Сохранение файла

5. Прделаем с текстом стандартные процедуры редактирования. Каждое действие будет осуществляться комбинацией клавиш.
 - 5.1. Вырезать одной командой целую строку (C-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-/).



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

Рис. 3.5: Текст после изменений

6. Научимся использовать команды по перемещению курсора. 6.1 . Переместим курсор в начало строки (C-a).

6.2 . Переместим курсор в конец строки (C-e).

6.3 . Переместим курсор в начало буфера (M-<).

6.4 . Переместим курсор в конец буфера (M->).

7. Управление буферами.

7.1. Выведем список активных буферов на экран (C-x C-b).

-:*** lab07.sh All L12 (Shell-script[bash])				
CRM	Buffer	Size	Mode	File
	* lab07.sh	151	Shell-script[ba...	~/lab07.sh
%*	*Messages*	1208	Messages	
%	*GNU Emacs*	727	Fundamental	
	scratch	145	Lisp Interaction	

Рис. 3.6: Список активных буферов

7.2. Переместимся во вновь открытое окно (C-x) со списком открытых буферов и переключимся на другой буфер.

```

-:*** lab07.sh All L12 (Shell-script[bash])
Mark set [4 times]
next-line: End of buffer
Undo
xref-pop-marker-stack: Marker stack is empty
exchange-point-and-mark: No mark set in this buffer
goto-history-element: End of defaults; no next item
goto-history-element: Beginning of history; no preceding item

U:*** *Messages* Bot L23 (Messages)
Visit tags table (default TAGS): ~/lab07.sh

```

Рис. 3.7: Переключение на другой буфер

7.3. Закроем это окно (C-x 0).

7.4. Теперь вновь переключимся между буферами, но уже без вывода их списка на экран (C-x b).

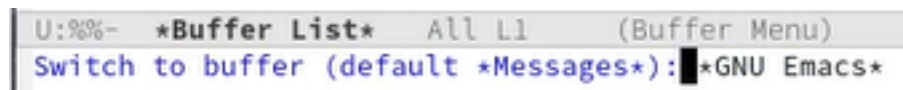


Рис. 3.8: Переключение между буферами

8. Управление окнами.

8.1. Поделим фрейм на 4 части: разделим фрейм на два окна по вертикали (С-х 3), а затем каждое из этих окон на две части по горизонтали (С-х 2).

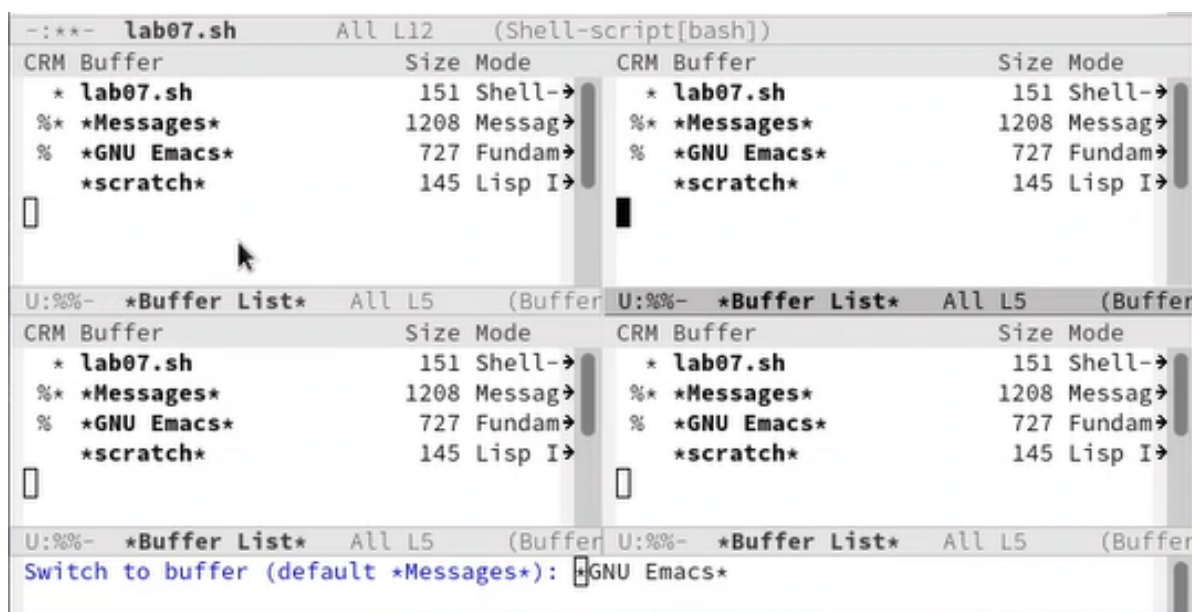


Рис. 3.9: Деление фрейма на четыре части

8.2. В каждом из четырёх созданных окон откроем новый буфер (файл) и введём несколько строк текста.

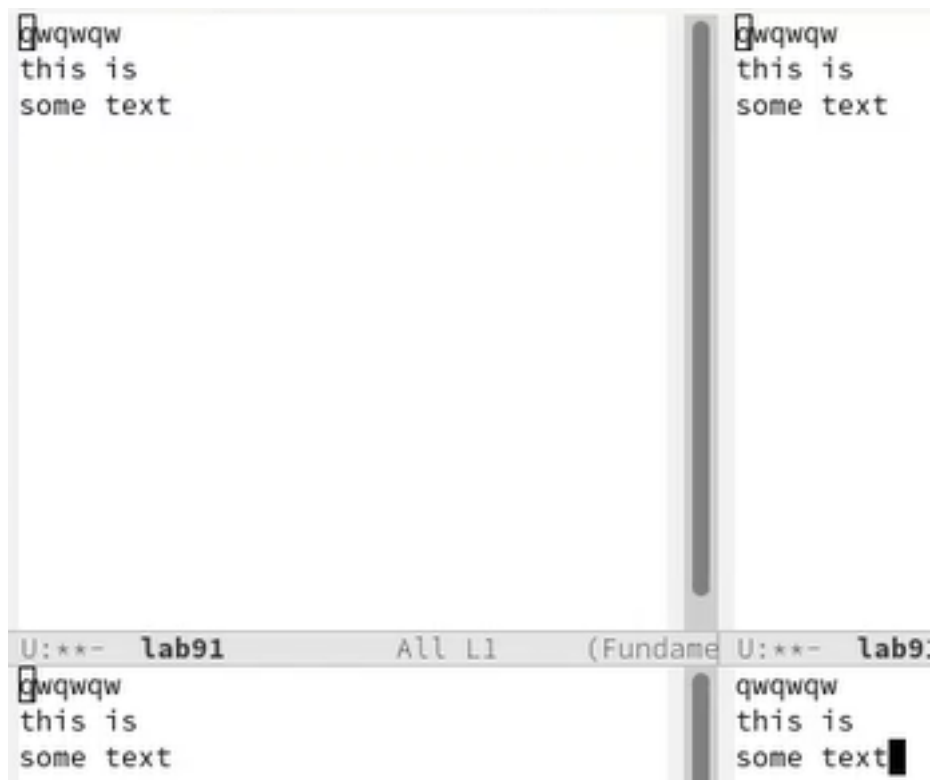


Рис. 3.10: Текст в буфере в четырёх окнах

9. Режим поиска.

9.1. Переключимся в режим поиска (C-s) и найдём несколько слов, присутствующих в тексте.

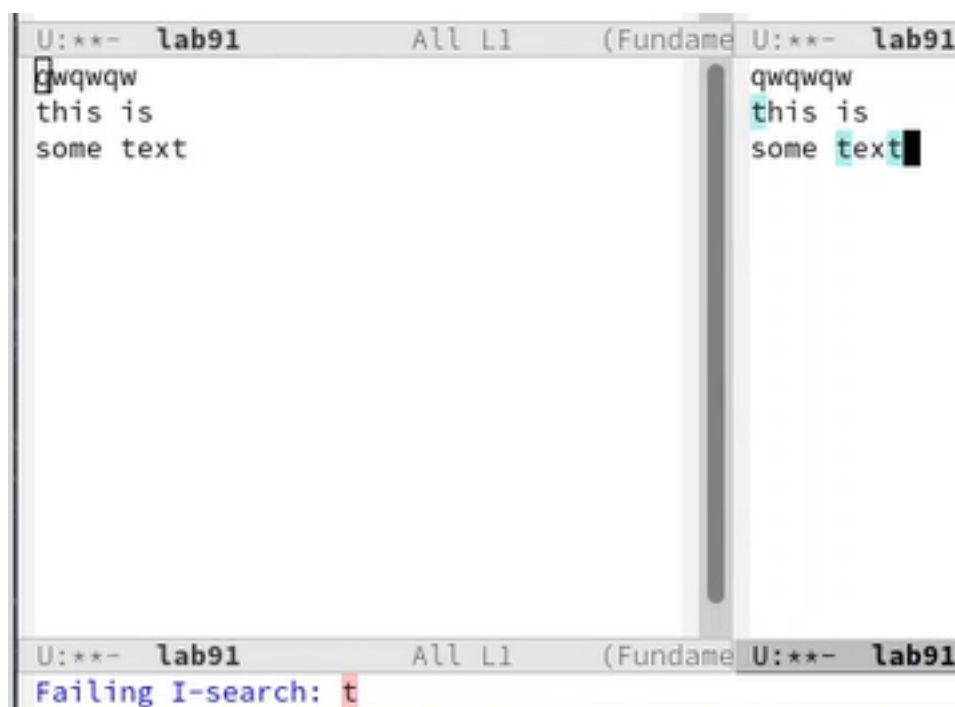


Рис. 3.11: Поиск слов в тексте

9.2. Переключаемся между результатами поиска, нажимая C-s.

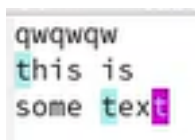


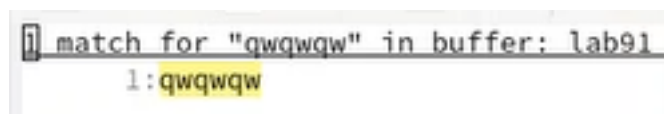
Рис. 3.12: Переключение между результатами поиска

9.3. Выйдем из режима поиска, нажав C-g.

9.4. Перейдём в режим поиска и замены (M-%), введём текст, который следует найти и заменить, нажмём Enter, затем введём текст для замены. После того как будут подсвечены результаты поиска, нажмём ! для подтверждения замены.

9.5. Попробуем другой режим поиска, нажав M-s o. Этот режим поиска отличается от предыдущего выводом более подробной информации, а именно

указанием имени файла, в котором находится это слово, и номера строки.



```
1 match for "qwqwqw" in buffer: lab91
1: qwqwqw
```

Рис. 3.13: Результат команды M-s o

4 Контрольные вопросы

1. Кратко охарактеризуйте редактор emacs.

Emacs - это 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. Какие особенности данного редактора могут сделать его сложным для освоения новичком?

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 придётся часто обращаться к справочным материалам.

3. Своими словами опишите, что такое буфер и окно в терминологии emacs'а.

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’a 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. Можно ли открыть больше 10 буферов в одном окне?

Да, можно.

5. Какие буферы создаются по умолчанию при запуске emacs?

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. Какие клавиши вы нажмёте, чтобы ввести следующую комбинацию C-c | и C-c C- |?

C-c | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и нажимаю «|» C-c C- | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и, удерживая «ctrl», нажимаю «|».

7. Как поделить текущее окно на две части?

Чтобы поделить окно на две части необходимо воспользоваться комбинацией «Ctrl-x 3» (по вертикали) или «Ctrl-x 2» (по горизонтали).

8. В каком файле хранятся настройки редактора emacs?

Настройки Emacs хранятся в файле .emacs.

9. Какую функцию выполняет клавиша Стрелка Влево и можно ли её переназначить?

По умолчанию клавиша «Стрелка Влево» удаляет символ перед курсором, но в редакторе её можно переназначить. Для этого необходимо изменить конфигурацию файла .emacs.

10. Какой редактор вам показался удобнее в работе vi или emacs? Поясните почему.

Более удобным я считаю редактор emacs, потому что в нём 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.

5 Выводы

Ознакомлены с операционной системой Linux. Получены практические навыки работы с редактором Emacs.