

Отчет по лабораторной работе № 11

Текстовый редактор etacs

Скобеева Алиса Алексеевна

Содержание

| | | |
|----------|---------------------------------------|-----------|
| 1 | Цель работы | 5 |
| 2 | Задание | 6 |
| 3 | Выполнение лабораторной работы | 7 |
| 4 | Выводы | 10 |
| 5 | Ответы на контрольные вопросы | 11 |

Список иллюстраций

| | | |
|-----|--|---|
| 3.1 | Набор текста в файл | 7 |
| 3.2 | Иллюстрация открытых буферов | 8 |
| 3.3 | Управление окнами | 8 |
| 3.4 | Набор текста в файл | 9 |
| 3.5 | Режим поиска | 9 |

Список таблиц

1 Цель работы

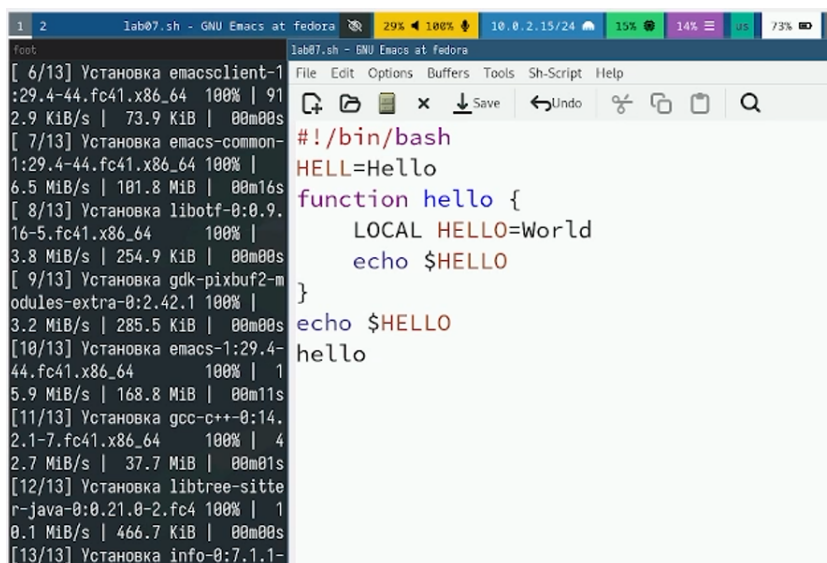
Получить практические навыки работы с редактором Emacs.

2 Задание

Ознакомиться с редактором Emacs и выполнить упражнения.

3 Выполнение лабораторной работы

Открываем Emacs и создаем файл lab07.sh с помощью комбинации клавиш. Набираем в нем текст.



```
1 2 lab07.sh - GNU Emacs at Fedora 29% 100% 10.0.2.15/24 15% 14% 73%
root [ 6/13] Установка emacsclient-1
:29.4-44.fc41.x86_64 100% | 91
2.9 KiB/s | 73.9 KiB | 00m00s
[ 7/13] Установка emacs-common-
1:29.4-44.fc41.x86_64 100% |
6.5 MiB/s | 101.8 MiB | 00m16s
[ 8/13] Установка libotf-0:0.9.
16-5.fc41.x86_64 100% |
3.8 MiB/s | 254.9 KiB | 00m00s
[ 9/13] Установка gdk-pixbuf2-m
odules-extra-0:2.42.1 100% |
3.2 MiB/s | 285.5 KiB | 00m00s
[10/13] Установка emacs-1:29.4-
44.fc41.x86_64 100% | 1
5.9 MiB/s | 168.8 MiB | 00m11s
[11/13] Установка gcc-c++-0:14.
2.1-7.fc41.x86_64 100% | 4
2.7 MiB/s | 37.7 MiB | 00m01s
[12/13] Установка libtree-sitte
r-java-0:0.21.0-2.fc4 100% | 1
0.1 MiB/s | 466.7 KiB | 00m00s
[13/13] Установка info-0:7.1.1-
lab07.sh - GNU Emacs at Fedora
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
```

Рис. 3.1: Набор текста в файл

Проделываем с текстом различные процедуры редактирования, используя комбинации клавиш.

Выполняем команды для перемещения курсора по тексту.

Выводим список активных буферов на экран. Переключаемся между буферами.

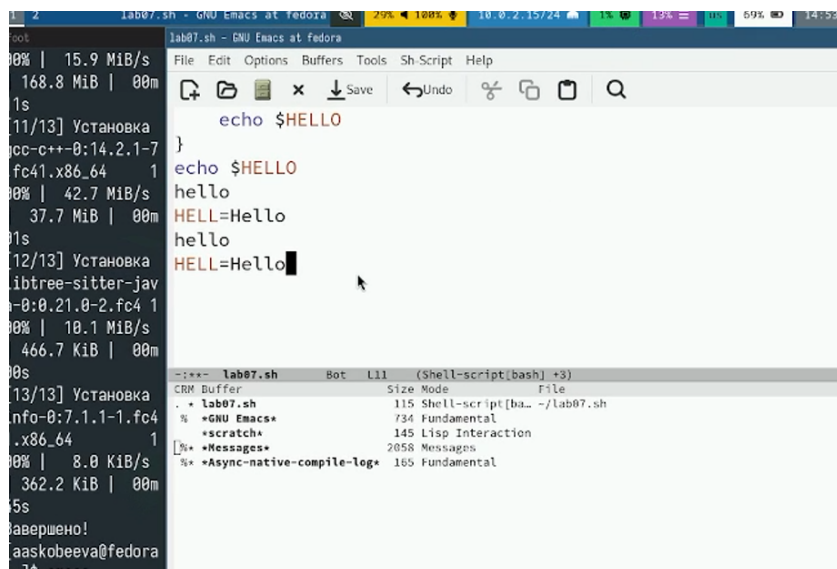


Рис. 3.2: Иллюстрация открытых буферов

Управление окнами. Делим фрейм на 4 части.

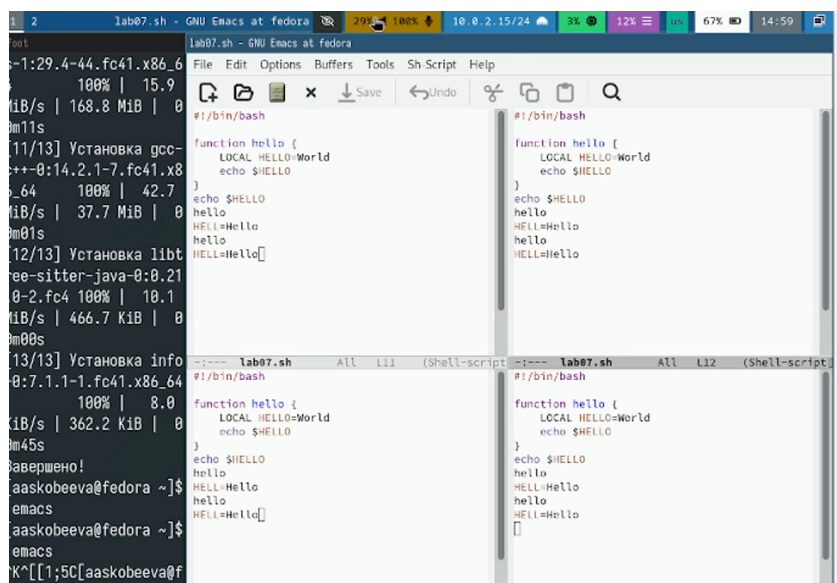


Рис. 3.3: Управление окнами

В каждом из четырех окон открываем новый буфер и вводим несколько строк текста.

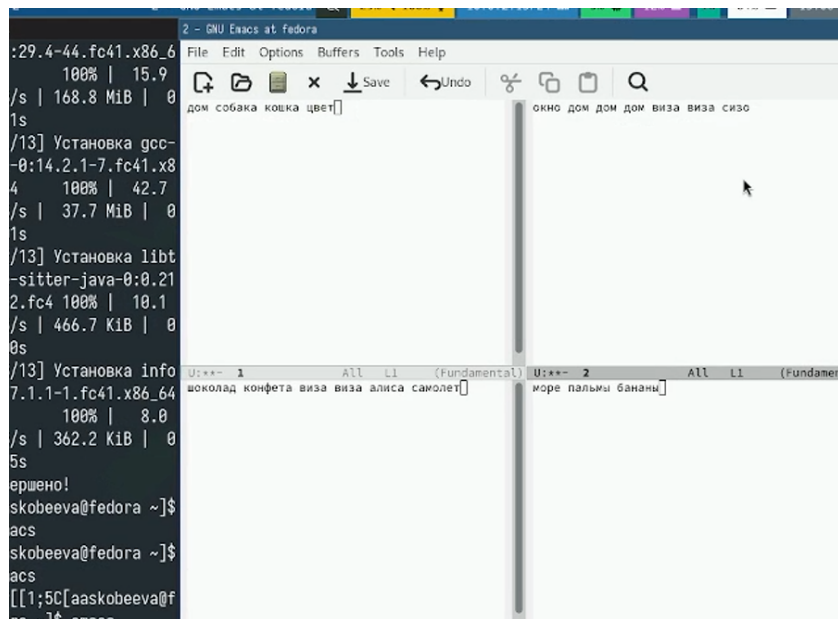


Рис. 3.4: Набор текста в файл

Пробуем несколько режимов поиска, и ищем слова по запросу в каждом из окон.

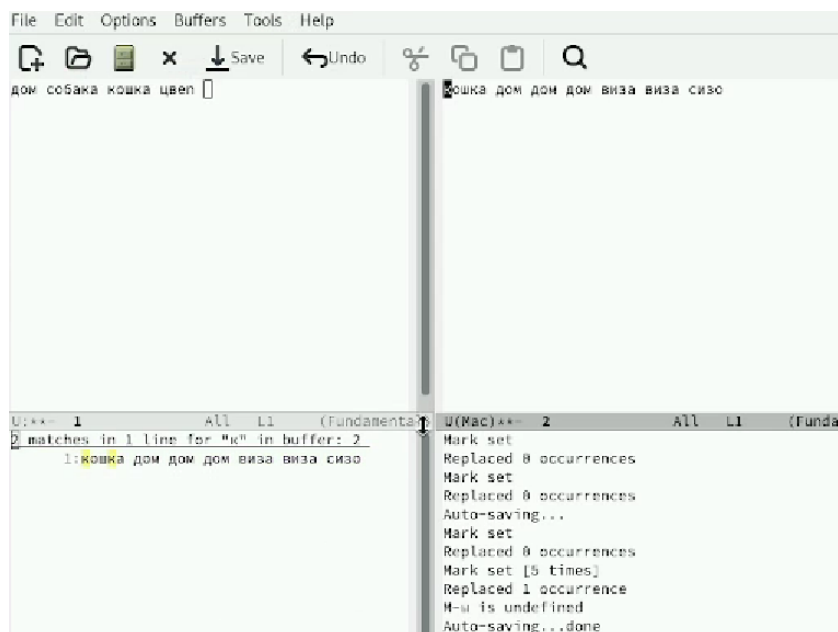


Рис. 3.5: Режим поиска

4 Выводы

Мы научились работать с текстовым редактором Emacs и успешно выполнили все задания лабораторной работы.

5 Ответы на контрольные вопросы

1. Краткая характеристика редактора Emacs: Emacs - это не просто текстовый редактор, а целая операционная среда внутри другой операционной системы. Это крайне мощный и гибкий инструмент, который позволяет редактировать текстовые файлы, писать код, управлять файлами, читать почту, просматривать веб-страницы и многое другое. Emacs славится своей расширяемостью: его можно настраивать и модифицировать практически бесконечно, используя язык Emacs Lisp. Это делает Emacs одним из самых настраиваемых редакторов, позволяя адаптировать его под конкретные нужды пользователя. Многие программисты считают Emacs своей “операционной системой”.
2. Особенности, затрудняющие освоение:
 - Кривая обучения: Emacs имеет довольно крутую кривую обучения. Огромное количество команд и комбинаций клавиш поначалу кажутся неподъемными.
 - Сложные комбинации клавиш: Многие команды требуют удержания нескольких клавиш одновременно, что может быть неудобно. Например, использование клавиши Meta (обычно Alt) в сочетании с другими клавишами.
 - “Все есть буфер”: Концепция буферов, где даже команды, сообщения и вывод программ представляются в виде буферов, может быть неочевидной для новичков.
 - Необходимость кастомизации: Чтобы Emacs стал действительно удобным, его нужно настраивать. Это требует знания Emacs Lisp, что является дополнительным барьером.
 - Ориентация на текст: Emacs в большей степени ориентирован на работу с текстом, а не на графический интерфейс, что мо-

- жет быть непривычно для тех, кто привык к современным GUI-редакторам.
3. Буфер и окно в Emacs: • Буфер: Представьте буфер как область памяти, содержащую текст, с которым вы работаете. Это может быть содержимое файла, вывод какой-то команды, список открытых файлов, черновик письма и т.д. Буфер существует независимо от того, отображается он на экране или нет. В Emacs практически все действия выполняются в буферах. • Окно: Окно - это область на экране, которая отображает содержимое одного из буферов. Можно сказать, что окно - это “вид” на буфер. Вы можете иметь несколько окон, каждое из которых отображает свой буфер, или несколько окон, отображающих один и тот же буфер.
 4. Более 10 буферов в одном окне? Нет, в каждый момент времени одно окно Emacs может отображать содержимое только одного буфера. Однако, вы можете быстро переключаться между разными буферами в одном окне, или создать несколько окон и в каждом открыть свой буфер. Таким образом, общее количество открытых буферов может быть сколь угодно большим.
 5. Буферы, создаваемые по умолчанию: • scratch: Буфер, предназначенный для экспериментов с Emacs Lisp. В нем можно выполнять небольшие фрагменты кода и сразу видеть результат. • Messages: Буфер, в котором отображаются системные сообщения, ошибки, предупреждения и другая информация о работе Emacs.
 6. Комбинации клавиш: • C-c | (Control+c, Shift+): Нажимаете и удерживаете клавишу Control, затем нажимаете клавишу ‘c’, отпускаете Control, затем нажимаете клавишу Shift и клавишу, на которой изображен символ | (обычно находится над Enter). • C-c C-| (Control+c, Control+Shift+): Нажимаете и удерживаете клавишу Control, затем нажимаете клавишу ‘c’, отпускаете Control, затем нажимаете и удерживаете клавишу Control и Shift, затем нажимаете клавишу, на которой изображен символ | (обычно находится над Enter).
 7. Разделить текущее окно на две части: • C-x 2 (Control+x, 2): Делит текущее окно по горизонтали на две части, в каждой из которых можно отобразить

свой буфер. • C-x 3 (Control+x, 3): Делит текущее окно по вертикали на две части.

8. Файл настроек: Emacs хранит свои настройки в файле, который обычно называется `~/.emacs` или `~/.emacs.d/init.el`. Это файл на языке Emacs Lisp, в котором можно указывать параметры настройки, определять новые команды и выполнять другие действия клавиша “стрелка влево” перемещает курсор на один символ влево. Да, её можно переназначить, используя Emacs Lisp. Например, можно назначить ей выполнение какой-либо другой команды или функции.
9. Удобство (субъективно): Личное мнение о удобстве работы с vi или Emacs сильно зависит от привычек, задач и предпочтений.

- Emacs: Мне лично Emacs кажется более удобным из-за его гибкости и расширяемости. Возможность настроить его под свои конкретные нужды и использовать для решения самых разных задач делает его незаменимым инструментом. Однако, я признаю, что освоение Emacs требует времени и усилий.
- vi: vi хорош своей лаконичностью и доступностью. Он практически всегда есть на любой Unix-подобной системе, и его базовые команды можно освоить за несколько минут. Это делает vi отличным выбором для быстрой правки файлов на удаленных серверах.