

Лабораторная работа №11

Текстовый редактор etacs

Юсупова Ксения Равиловна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	15
4	Ответы на контрольные вопросы	16

Список иллюстраций

2.1	Открыли etags	6
2.2	Создали файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f и набрали текст	7
2.3	Вырезали строку и вставили в конец файла	7
2.4	Выделили область текста и скопировали	8
2.5	Вставили область текста в конец файла	8
2.6	Вновь выделили эту область и на этот раз вырезали её	8
2.7	Отменили последнее действие	9
2.8	Вывели список активных буферов	9
2.9	Переместились в открытое окно со списком открытых буферов и переключились на другой буфер	10
2.10	Вновь переключились между буферами	11
2.11	Поделили фрейм на 4 части и в каждом из четырёх созданных окон открыли новый буфер (файл)	12
2.12	Перешли в режим поиска и замены (M-%), ввели текст, который следует найти	13
2.13	Заменили текст	13
2.14	Испробовали другой режим поиска, нажав M-s	14

Список таблиц

1 Цель работы

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

2 Выполнение лабораторной работы

Открыли emacs(рис. 2.1).

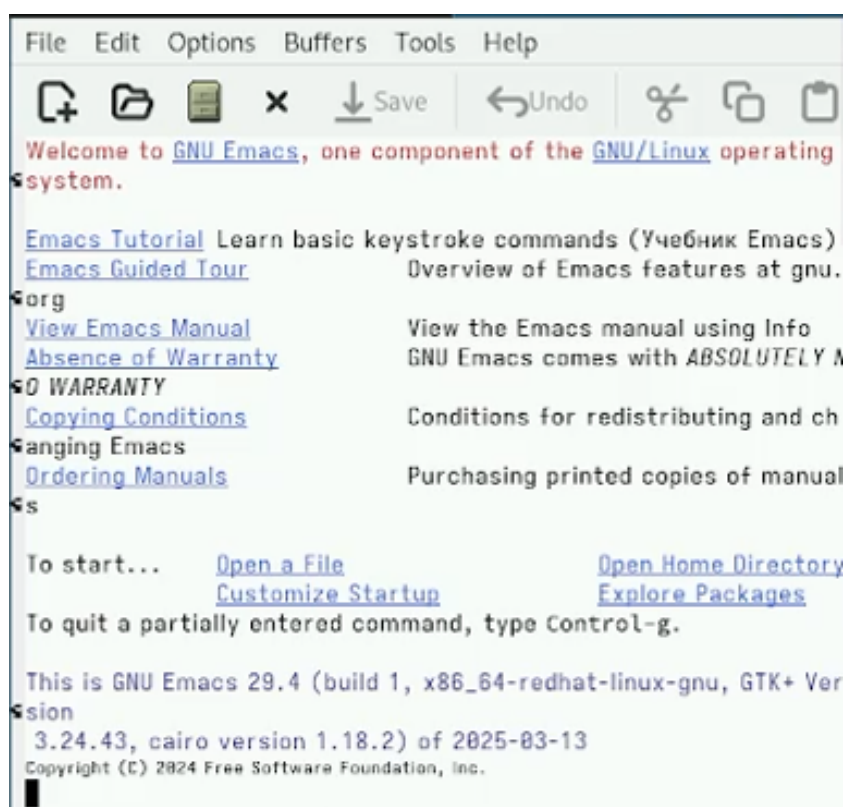


Рис. 2.1: Открыли emacs

Создали файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f и набрали текст(рис. 2.2).

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

Рис. 2.2: Создали файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f и набрали текст

Вырезали одной командой целую строку (C-k) и вставили эту строку в конец файла (C-y)(рис. 2.3).



```
RUU:~ - lab07.sh All L10 (Shell-script[sh])
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}
hello
echo $HELLO
```

Рис. 2.3: Вырезали строку и вставили в конец файла

Выделили область текста (C-space) и скопировали область в буфер обмена (M-w).(рис. 2.4).

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

hello
echo $HELLO
```

Рис. 2.4: Выделили область текста и скопировали

Вставили область текста в конец файла(рис. 2.5).

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

hello
echo $HELLO
LOCAL HELLO=World
echo $HELLO
```

Рис. 2.5: Вставили область текста в конец файла

Вновь выделили эту область и на этот раз вырезали её (C-w)(рис. 2.6).

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

hello
echo $HELLO
```

Рис. 2.6: Вновь выделили эту область и на этот раз вырезали её

Отменили последнее действие (C-/).(рис. 2.7).


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

hello
echo $HELLO
LOCAL HELLO=World
echo $HELLO
```

Рис. 2.7: Отменили последнее действие

Научились использовать команды по перемещению курсора. Переместили курсор в начало строки, в конец строки, в начало буфера, в конец буфера.

Вывели список активных буферов на экран (C-x C-b).(рис. 2.8).

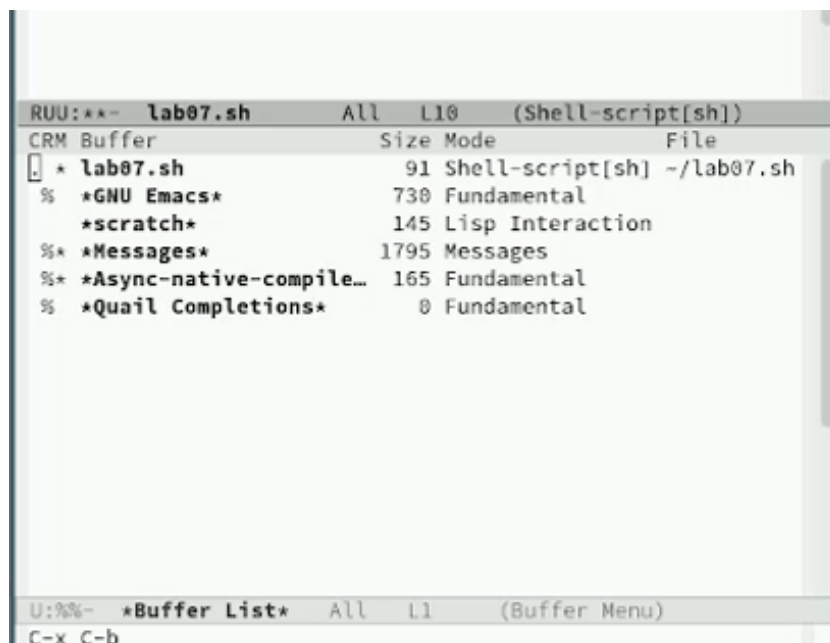


Рис. 2.8: Вывели список активных буферов

Переместились во вновь открытое окно (C-x) о со списком открытых буферов и переключились на другой буфер.(рис. 2.9).



```
RUU:~ - lab07.sh All L10 (Shell-script[sh])
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}

hello
echo $HELLO
```

Рис. 2.9: Переместились в открытое окно со списком открытых буферов и переключились на другой буфер

Теперь вновь переключился между буферами, но уже без вывода их списка на экран (C-x b)(рис. 2.10).

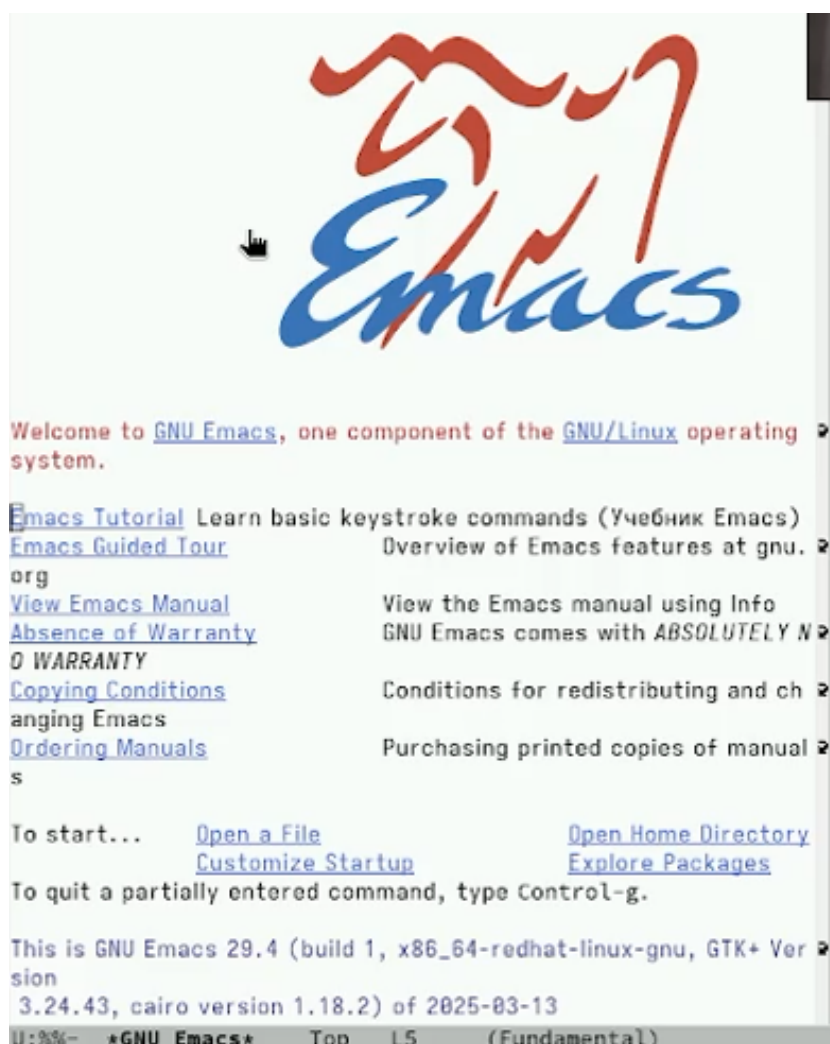


Рис. 2.10: Вновь переключились между буферами

Поделили фрейм на 4 части: разделили фрейм на два окна по вертикали (С-х 3), а затем каждое из этих окон на две части по горизонтали (С-х 2). В каждом из четырёх созданных окон открыли новый буфер (файл) и ввели несколько строк текста(рис. 2.11).



Рис. 2.11: Поделили фрейм на 4 части и в каждом из четырёх созданных окон открыли новый буфер (файл)

Перешли в режим поиска и замены (M-%), ввели текст, который следует найти (рис. 2.12).

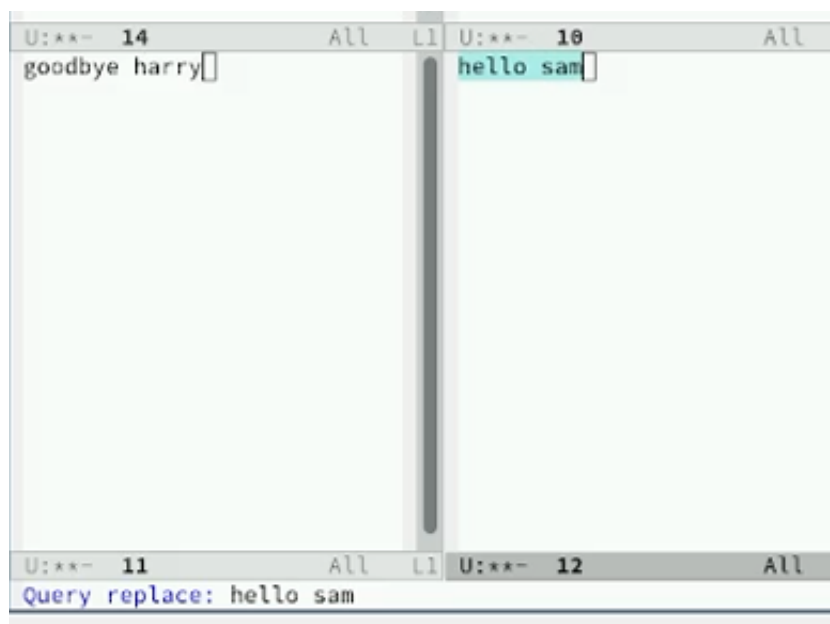


Рис. 2.12: Перешли в режим поиска и замены (М-%), ввели текст, который следует найти

Заменяли текст(рис. 2.13).



Рис. 2.13: Заменяли текст

Испробовали другой режим поиска, нажав М-s (рис. 2.14).



Рис. 2.14: Испробовали другой режим поиска, нажав M-s

3 Выводы

В ходе лабораторной работы больше познакомились с операционной системой Linux. Получили практические навыки работы с редактором Emacs.

4 Ответы на контрольные вопросы

1. Краткая характеристика редактора Emacs

Emacs — расширяемый кроссплатформенный текстовый редактор, работающий в терминале и графическом режиме. В Fedora устанавливается пакетом emacs или emacs-nox (консольная версия). Поддерживает:

- Гибкую работу в терминале под Sway/Wayland
- Полную интеграцию с shell-окружением
- Управление через клавиатуру (важно для tiling WM)

2. Особенности, осложняющие освоение в Sway

- Комбинации клавиш могут конфликтовать с хоткеями Sway (например, Mod+C)
- Отсутствие графического меню по умолчанию
- Сложность переключения между буферами без мыши

3. Буфер и окно в терминологии Emacs

- **Буфер** - аналог вкладки (файл, процесс, лог)
- **Окно** - область отображения (в терминале - разделённые tmux-подобные области)

4. Можно ли открыть 10 буферов в одном окне?

Да, ограничений нет.

5. Стандартные буферы при запуске:

- `*scratch*`
- `*Messages*`
- `*GNU Emacs*`

6. Ввод комбинаций:

- `C-c |`: `Ctrl+c`, затем `Shift+`
- `C-c C- |`: `Ctrl+c`, затем (с зажатым `Ctrl`) `Shift+`

7. Разделение окна:

- Горизонтально: `C-x 2`
- Вертикально: `C-x 3`

8. Файлы настроек:

- `~/.emacs.d/init.el`
- `~/.config/emacs/early-init.el`

9. Клавиша `Esc`:

- Выполняет функцию Meta-клавиши
- Можно переназначить через `(global-set-key)`

10. **Vi** предпочтительнее в Linux Fedora со Sway, потому что:

- Имеет более эргономичные сочетания клавиш, не конфликтующие с хоткеями Sway

- Быстрее запускается и работает в терминальном режиме
- Лучше интегрируется с tiling window manager (нативное разделение окон)
- Требуется меньше ресурсов для базового использования
- Более стандартизированная навигация (hjkl)
- Удобнее для администрирования системы и быстрого редактирования конфигов
- Имеет предустановленную версию (vi) во всех дистрибутивах Linux
- Лучше подходит для работы через SSH