

# **Лабораторная работа №11**

**Текстовый редактор etacs**

Полякова Юлия Александровна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>4</b>	<b>Контрольные вопросы</b>	<b>15</b>
<b>5</b>	<b>Вывод</b>	<b>18</b>

# Список иллюстраций

3.1	Создание и заполнение файла . . . . .	7
3.2	Сохранение файла . . . . .	7
3.3	Результат процедур редактирования текста . . . . .	8
3.4	Перемещение курсора клавишами . . . . .	9
3.5	Список активных буферов . . . . .	9
3.6	Перемещаемся в открытое окно . . . . .	10
3.7	Другой буфер . . . . .	10
3.8	Закрытие окна . . . . .	11
3.9	Переключение без вывода списка . . . . .	11
3.10	Деление на 4 окна . . . . .	12
3.11	Режим поиска . . . . .	12
3.12	Переключение между результатами . . . . .	13
3.13	Поиск с заменой . . . . .	13
3.14	Иной режим поиска . . . . .	14

## **Список таблиц**

# 1 Цель работы

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

## **2 Задание**

1. Ознакомиться с теоретическим материалом.
2. Ознакомиться с редактором etas.
3. Выполнить упражнения.
4. Ответить на контрольные вопросы.

### 3 Выполнение лабораторной работы

1. Открываем emacs, написав в командной строке emacs. Создаем файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f). Набираем предложенный текст (рис. 3.1).

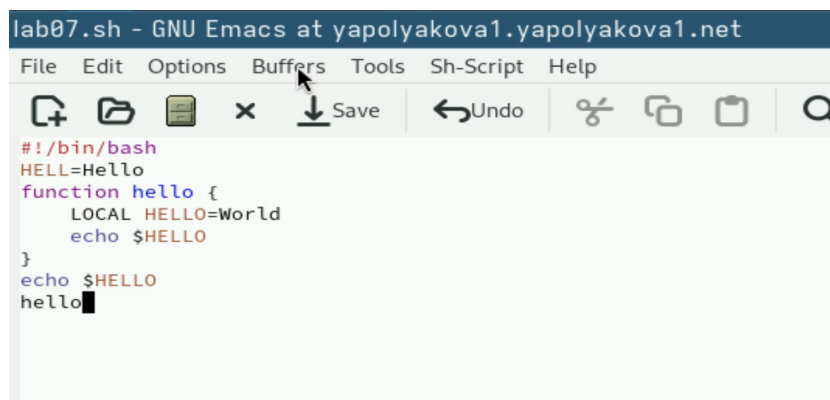


Рис. 3.1: Создание и заполнение файла

2. Сохраняем файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s). (рис. 3.2)

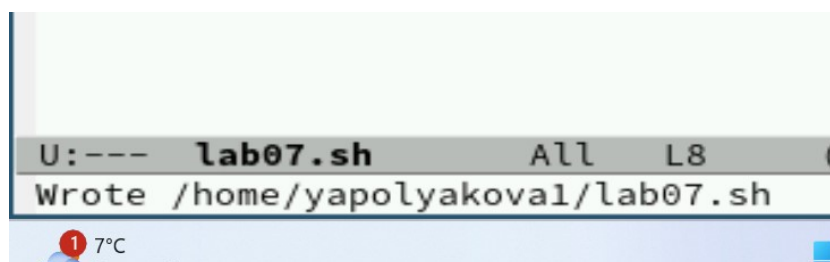
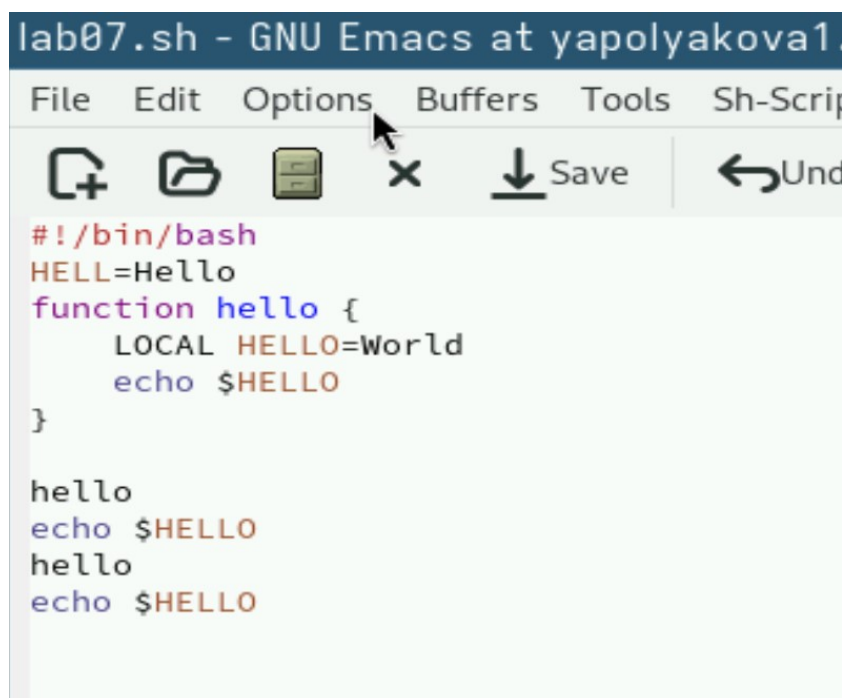


Рис. 3.2: Сохранение файла

3. Проделываем с текстом стандартные процедуры редактирования, каждое действие должно осуществляем комбинацией клавиш. Вырезаем одной

командой целую строку (C-k), я вырезала `echo $HELLO`. Вставляем эту строку в конец файла (C-y). Выделяем область текста (C-space и я использовала стрелки), я выделила две строки `hello` и `echo $HELLO`. Копируем область в буфер обмена (Alt-w). Вставляем область в конец файла. Вновь выделяем эту область и на этот раз вырезаем её (C-w). Отменяем последнее действие (C-/). Результат на рисунке (рис. 3.3)



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

hello
echo $HELLO
hello
echo $HELLO
```

Рис. 3.3: Результат процедур редактирования текста

4. Учимся использовать команды по перемещению курсора. Перемещаем курсор в начало строки (C-a). Перемещаем курсор в конец строки (C-e). Перемещаем курсор в начало буфера (Alt-<). Перемещаем курсор в конец буфера (Alt->). (рис. 3.4)



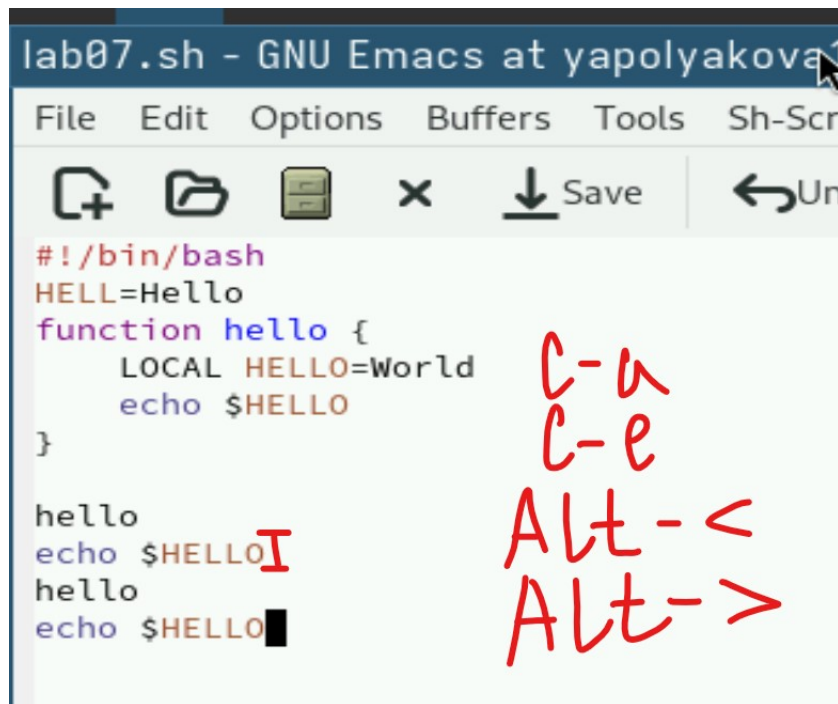


Рис. 3.4: Перемещение курсора клавишами

5. Выводим список активных буферов на экран (C-x C-b) (рис. 3.5)

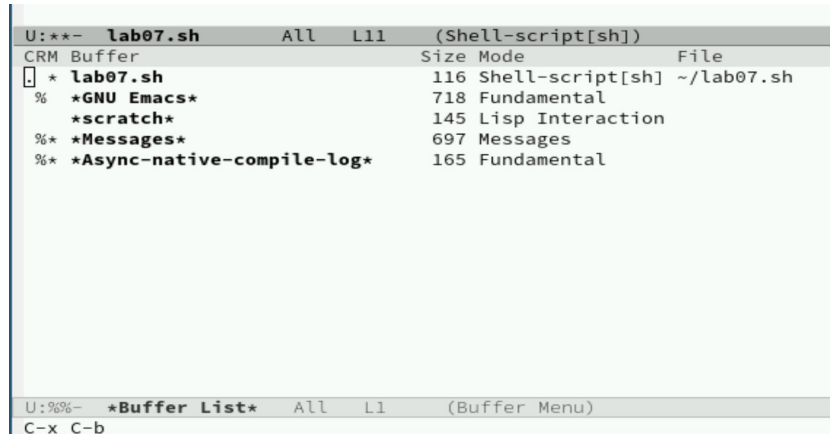


Рис. 3.5: Список активных буферов

6. Перемещаемся во вновь открытое окно (C-x o) со списком открытых буферов (рис. 3.6)

```
U:***- lab07.sh All L11 (Shell-script[sh])
CRM Buffer Size Mode File
* lab07.sh 116 Shell-script[sh] ~/lab07.sh
% *GNU Emacs* 718 Fundamental
*scratch* 145 Lisp Interaction
%* *Messages* 697 Messages
%* *Async-native-compile-log* 165 Fundamental

U:***- *Buffer List* All L1 (Buffer Menu)
C-x o
```

Рис. 3.6: Перемещаемся в открытое окно

7. Переключаемся на другой буфер (C-x b) (рис. 3.7)

```
U:***- lab07.sh All L11 (Shell-scrip
Wrote /home/yapolyakova1/lab07.sh
Undo
Mark set [2 times]
Mark activated
Mark set [2 times]
Undo
Mark set [2 times]
Auto-saving...

U:***- *Messages* Bot L19 (Messages)
```

Рис. 3.7: Другой буфер

8. Закрываем это окно (C-x 0) (рис. 3.8)



Рис. 3.8: Заккрытие окна

9. Переключаемся между буферами, но уже без вывода их списка на экран (C-x b). (рис. 3.9)

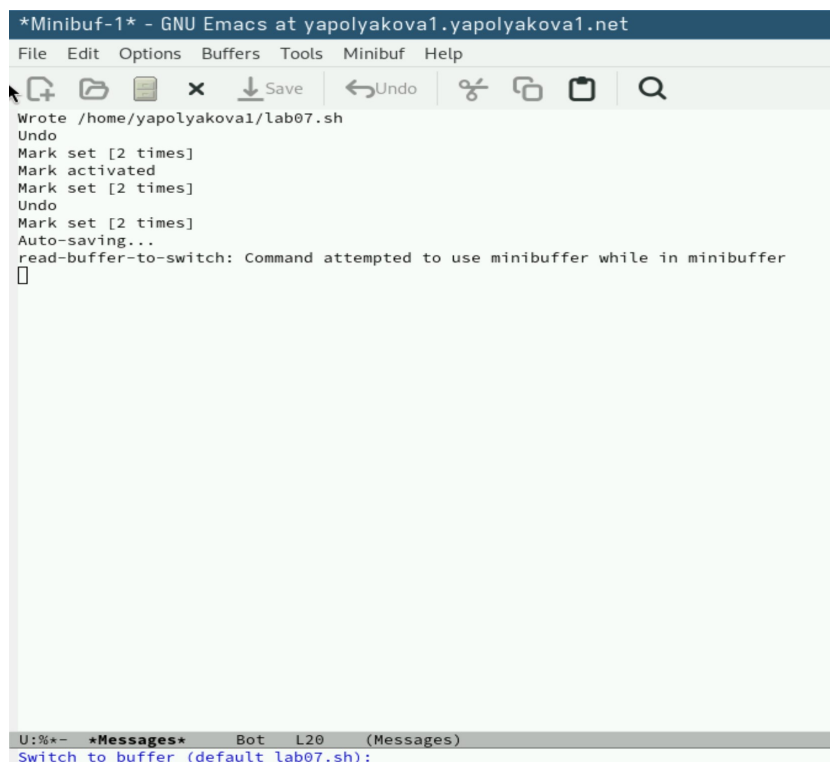


Рис. 3.9: Переключение без вывода списка

10. Делим фрейм на 4 части: делим на два окна по вертикали (С-х 3), а затем каждое из этих окон на две части по горизонтали (С-х 2) (переключаемся между окнами С-х о). В каждом из четырёх созданных окон открываем новый буфер (С-х С-f) и вводим несколько строк текста. (рис. 3.10)

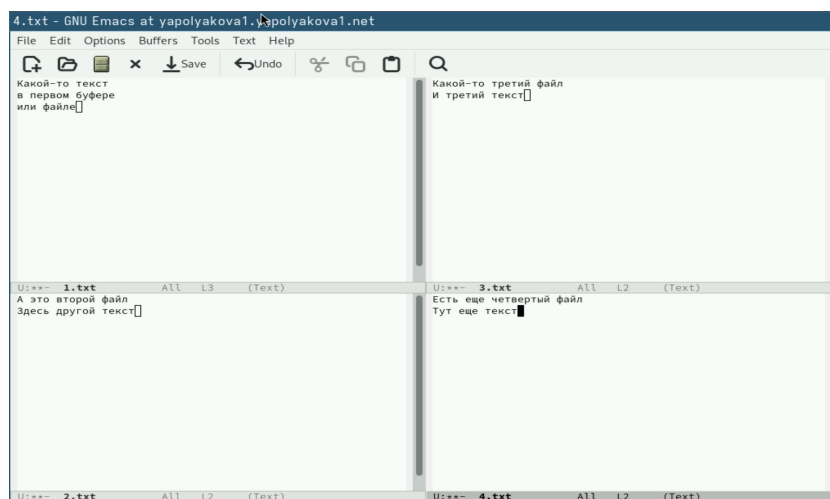


Рис. 3.10: Деление на 4 окна

11. Переключаемся в режим поиска (С-s) и ищем несколько слов, присутствующих в тексте. (рис. 3.11)



Рис. 3.11: Режим поиска

12. Переключаемся между результатами поиска, нажимая С-s. Выходим из режима поиска, нажав С-g (рис. 3.12)



Рис. 3.12: Переключение между результатами

13. Заменяем “еще” на “ещё”, используем режим поиска и замены (M-%), вводим текст, который следует найти и заменить, нажимаем Enter, затем вводим текст для замены. После того как будут подсвечены результаты поиска, нажимаем ! для подтверждения замены. (рис. 3.13)

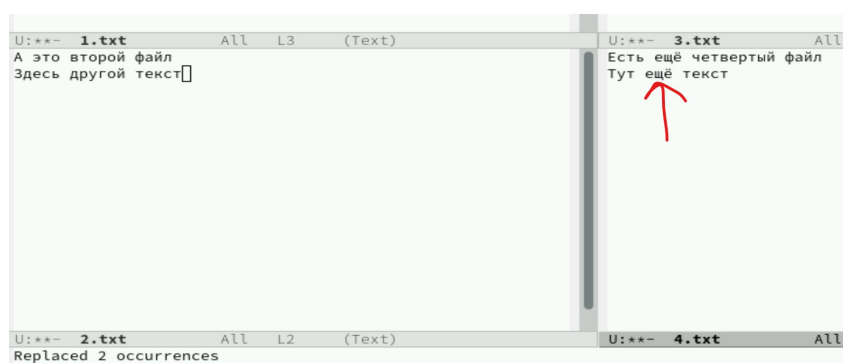


Рис. 3.13: Поиск с заменой

14. Пробуем другой режим поиска, нажав M-s o. Он отличается от предыдущих тем, что выводит результат поиска в отдельном окне. Он показывает кол-во совпадений, имя файла и место в файле (рис. 3.14)

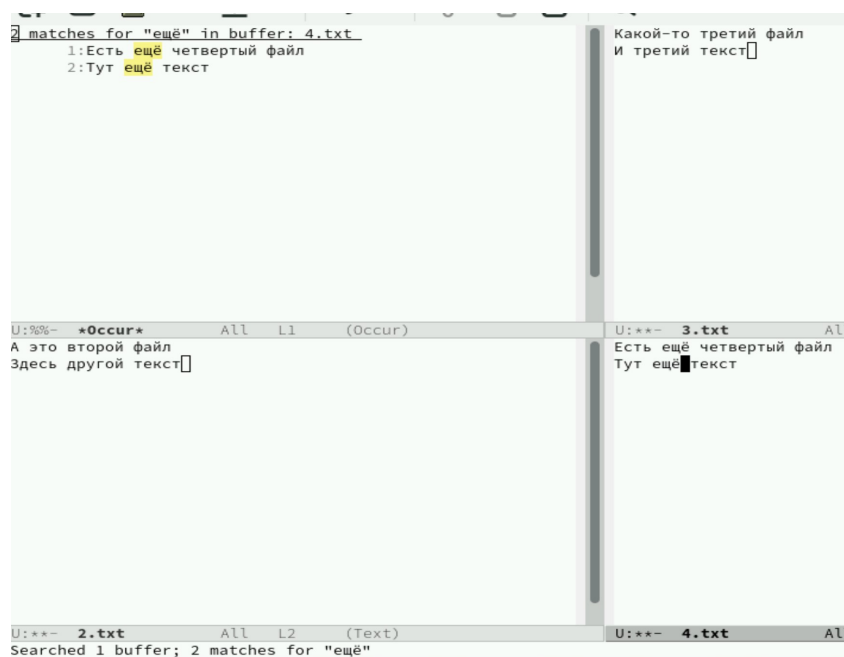


Рис. 3.14: Иной режим поиска

## 4 Контрольные вопросы

### 1. Краткая характеристика Emacs

Emacs — мощный расширяемый текстовый редактор с открытым исходным кодом, известный своей гибкостью и возможностями кастомизации. Поддерживает широкий спектр функций от простого редактирования текста до сложных операций программирования, организации рабочего процесса и даже электронной почты.

### 2. Особенности сложности для новичков:

- Многочисленные комбинации клавиш: основная функциональность требует запоминания множества сочетаний клавиш (C-x, C-s и др.).
- Строгая структура команд: некоторые команды требуют точного набора символов.
- Высокая степень настраиваемости: большое количество настроек и конфигураций делает начальное освоение непростой задачей.
- Необходимость изучения базовых концептов, таких как буферы, окна, режимы.

### 3. Буфер и окно в терминологии Emacs'a:

Буфер — это контейнер для хранения содержимого документа (текста), независимо от отображаемого окна. Окно же представляет собой область экрана, в которой отображается содержимое одного или нескольких буферов одновременно. Один буфер может отображаться сразу в нескольких окнах, а одно окно может показывать разные участки одного буфера.

#### 4. Можно ли открыть больше 10 буферов в одном окне?

Да, можно. Количество открытых буферов никак не ограничено количеством окон. Каждый буфер хранится отдельно, а окно лишь показывает определённый участок текущего активного буфера.

#### 5. Какие буферы создаются по умолчанию при запуске Emacs?

При старте Emacs автоматически создаёт два основных буфера:

- `*scratch*`: специальный буфер для временного ввода и тестирования выражений Lisp.
- `*Messages*`: здесь выводятся информационные сообщения и предупреждения системы.

#### 6. Комбинации клавиш:

Чтобы ввести последовательность C-c |, нажимаем сначала сочетание Ctrl+C, отпускаем, потом вводим символ вертикальной черты ("|"). Для последовательности C-c C-| выполняем аналогично, только вторую часть вводим удерживая Control и Shift: Ctrl+C → отпускаем → снова нажимаем Ctrl+Shift+"|".

#### 7. Разделение окна на две части:

Используя команду C-x 2 разделяет активное окно горизонтально пополам. Команда C-x 3 делит окно вертикально на две части.

#### 8. Файл конфигурации Emacs:

Настройки сохраняются в файле `.emacs.d/init.el`. Этот файл находится в домашнем каталоге пользователя и хранит всю необходимую конфигурацию и расширения.

#### 9. Клавиша Escape (или Meta):



Клавиша Esc (Meta) используется для запуска различных команд Emacs. Её часто применяют совместно с другими клавишами для активации специализированных действий. Эту клавишу можно переназначить путём изменения соответствующего макроса в настройках файла `init.el`.

#### 10. Удобство Emacs vs Vi/Vim:

Выбор между Emacs и Vim зависит от личных предпочтений и целей использования. Например, если важна высокая степень настройки и возможность автоматизации процессов разработки, Emacs станет лучшим выбором благодаря своей мощной экосистеме плагинов и встроенному языку сценариев (Elisp). Если важнее скорость навигации по тексту и минимальные усилия на обучение клавиатурных сокращений, то Vim может показаться предпочтительнее. Оба инструмента имеют сильные стороны, и выбор идеального решения индивидуален. Удобнее мне показался emacs, так как он наиболее приближен к привычным редакторам, например, к Блокноту или gedit.

## **5 Вывод**

Были получены практические навыки работы с редактором Emacs.