

# Лабораторная работа №9

Текстовый редактор emacs

Гнатюк Анастасия Станиславовна

# Содержание

Цель работы	3
Теоретическое введение	4
Последовательность выполнения работы	5
Выполнение лабораторной работы	6
Вывод	16
Контрольные вопросы	17

## Цель работы

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

# Теоретическое введение

Определение 1. 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.

Определение 2. Фрейм соответствует окну в обычном понимании этого слова. Каждый фрейм содержит область вывода и одно или несколько окон Emacs.

Определение 3. 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 C, режим Texinfo и другие. Под второстепенными режимами понимается список режимов, которые включены в данный момент в буфере выбранного окна.

Определение 4. Область вывода — одна или несколько строк внизу фрейма, в которой Emacs выводит различные сообщения, а также запрашивает подтверждения и дополнительную информацию от пользователя.

Определение 5. Минибуфер используется для ввода дополнительной информации и всегда отображается в области вывода.

Определение 6. Точка вставки — место вставки (удаления) данных в буфере.

# Последовательность выполнения работы

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

# Выполнение лабораторной работы

1. Открыть emacs.

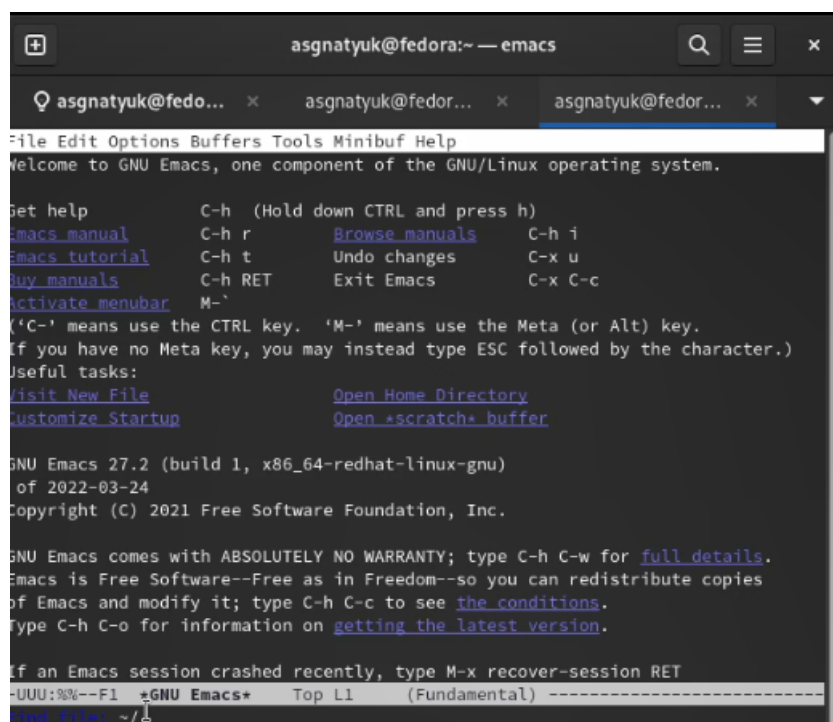


Рис. 1: Рис.1

2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).



Рис. 2: Рис.2

3. Наберите текст:

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

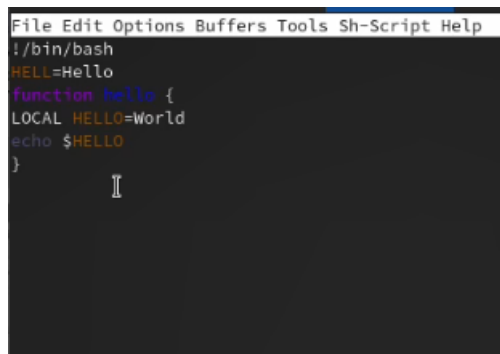


Рис. 3: Рис.3

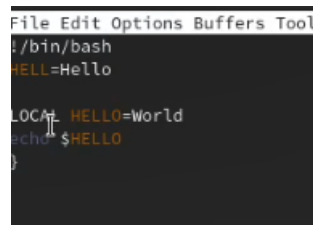
4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).

5. Прodelать с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш.

5.1. Вырезать одной командой целую строку (C-k).



Рис. 4: Рис.4

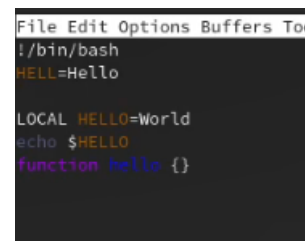


```
File Edit Options Buffers Tool
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}
```

Рис. 5: Рис.5

5.2. Вставить эту строку в конец файла (C-y).

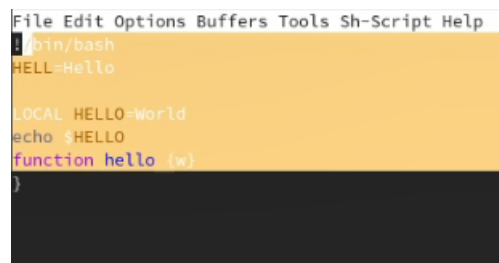


```
File Edit Options Buffers Too
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {}
```

Рис. 6: Рис.6

5.3. Выделить область текста (C-space).



```
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
}
```

Рис. 7: Рис.7

5.4. Скопировать область в буфер обмена (M-w).

5.5. Вставить область в конец файла.



```
File Edit Options Buffers Tools Sh-Script
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}

#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
```

Рис. 8: Рис.8

5.6. Вновь выделить эту область и на этот раз вырезать её (C-w).

```
File Edit Options Buffers Tools Sh-Script
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
}

#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
```


Рис. 9: Рис.9

```
File Edit Options Buffers Tools Sh-Script
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
}
```

Рис. 10: Рис.10

5.7. Отмените последнее действие (C-/).



```
File Edit Options Buffers Tools Sh-Script
/bin/bash
HELLO=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
}

/bin/bash
HELLO=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}

-UUU:***-F1 touch lab07.sh All L8
Undo
```

Рис. 11: Рис.11

6. Научитесь использовать команды по перемещению курсора.

6.1. Переместите курсор в начало строки (C-a).

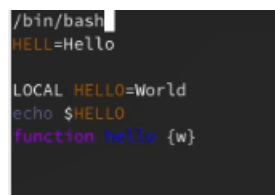


```
/bin/bash
HELLO=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
}
```

Рис. 12: Рис.12

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



```
/bin/bash
HELLO=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
}
```

Рис. 13: Рис.13

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

```
File Edit Options Buffers Tools Sh Script
/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}
}
/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
function hello {w}

-UUU:***F1 touch lab07.sh All L1
Beginning of buffer
```

Рис. 14: Рис.14

## 7. Управление буферами.

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

```
-UUU:***F1 touch lab07.sh Top L1 (Shell-script[sh]) -----
CRM Buffer      Size Mode      File
. * touch lab07.sh 145 Shell-script[sh] ~/touch lab07.sh
% *GNU Emacs* 1012 Fundamental
*scratch* 145 Lisp Interaction
%* *Messages* 1822 Messages

-UUU:%%F1 *Buffer List* All L1 (Buffer Menu) -----
C-x C-b
```

Рис. 15: Рис.15

7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых буферов и переключитесь на другой буфер.

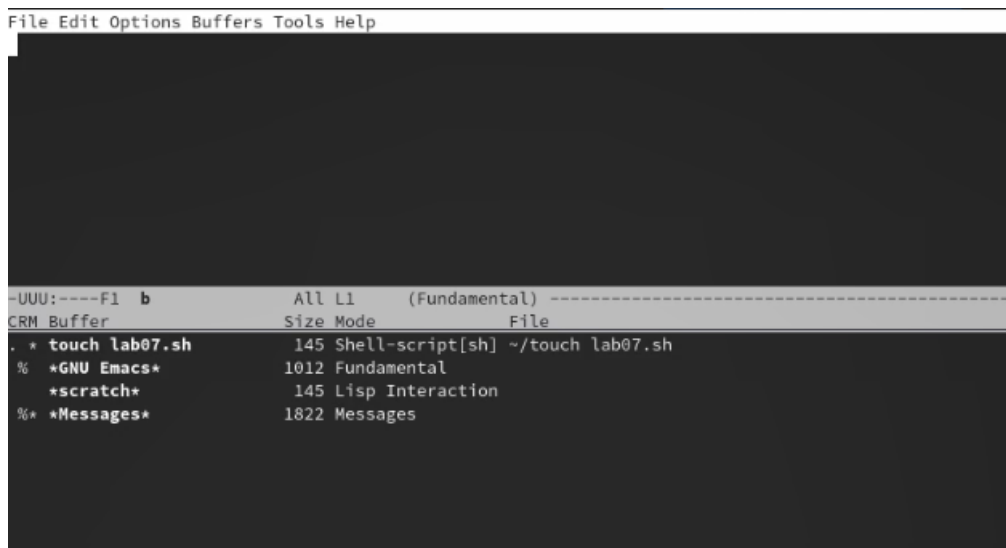


Рис. 16: Рис.16

7.3. Закройте это окно (C-x 0).

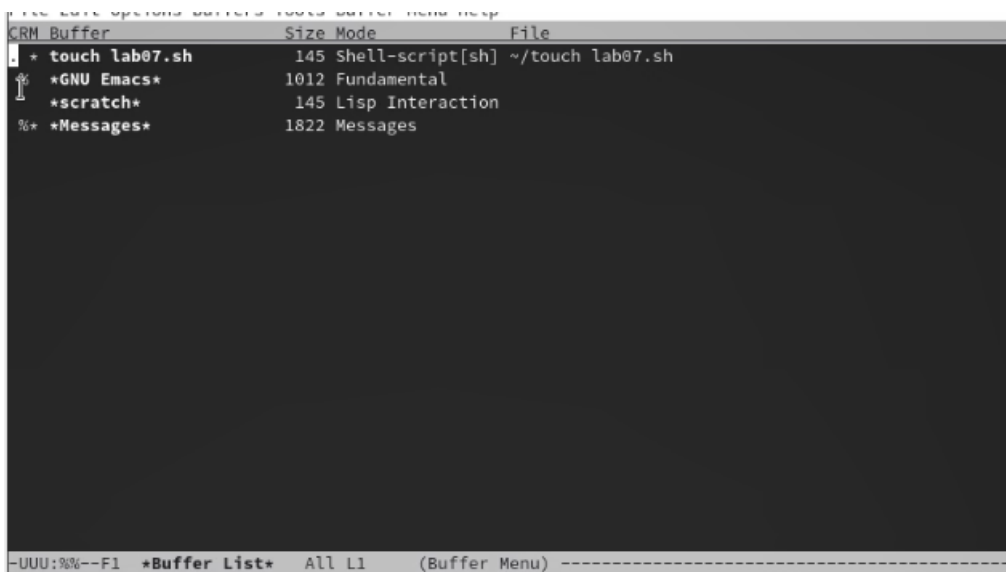


Рис. 17: Рис.17

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



Рис. 18: Рис.18

## 8. Управление окнами.

8.1. Поделите фрейм на 4 части: разделите фрейм на два окна по вертикали (С-х 3), а затем каждое из этих окон на две части по горизонтали (С-х 2) (см. рис. 9.1).

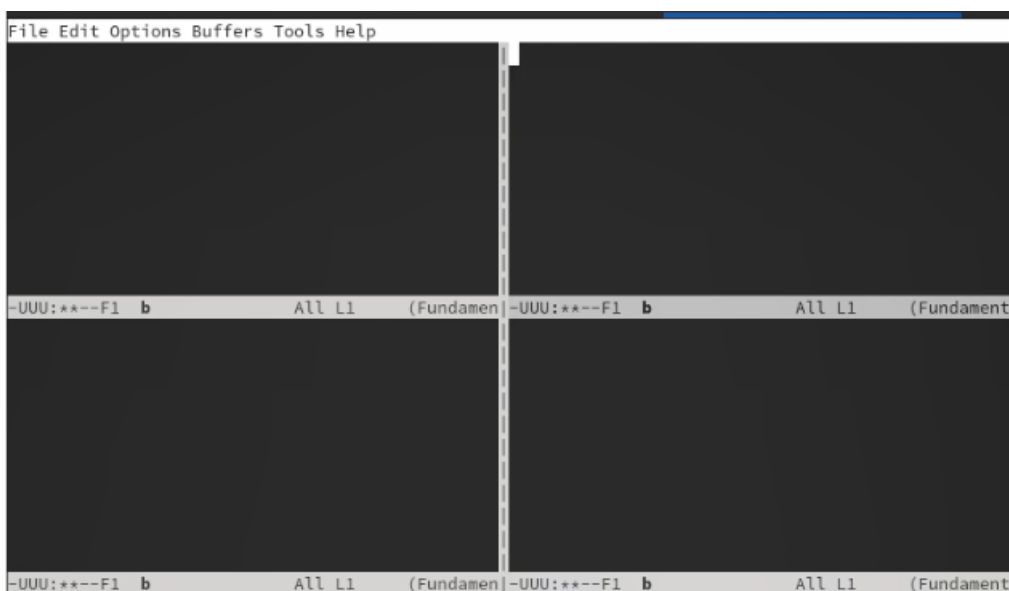


Рис. 19: Рис.19

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

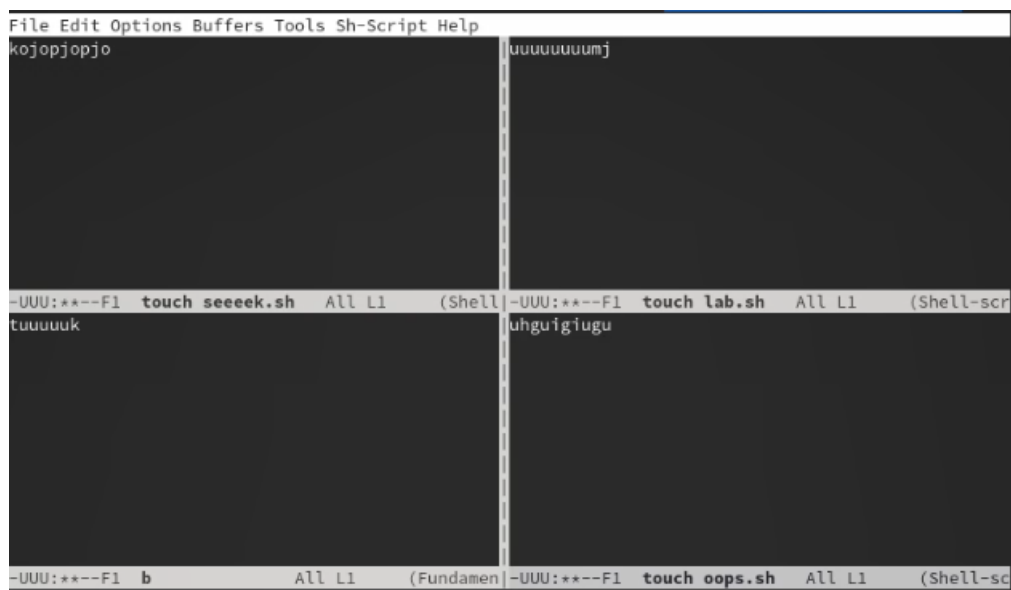


Рис. 20: Рис.20

## 9. Режим поиска

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



Рис. 21: Рис.21

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

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

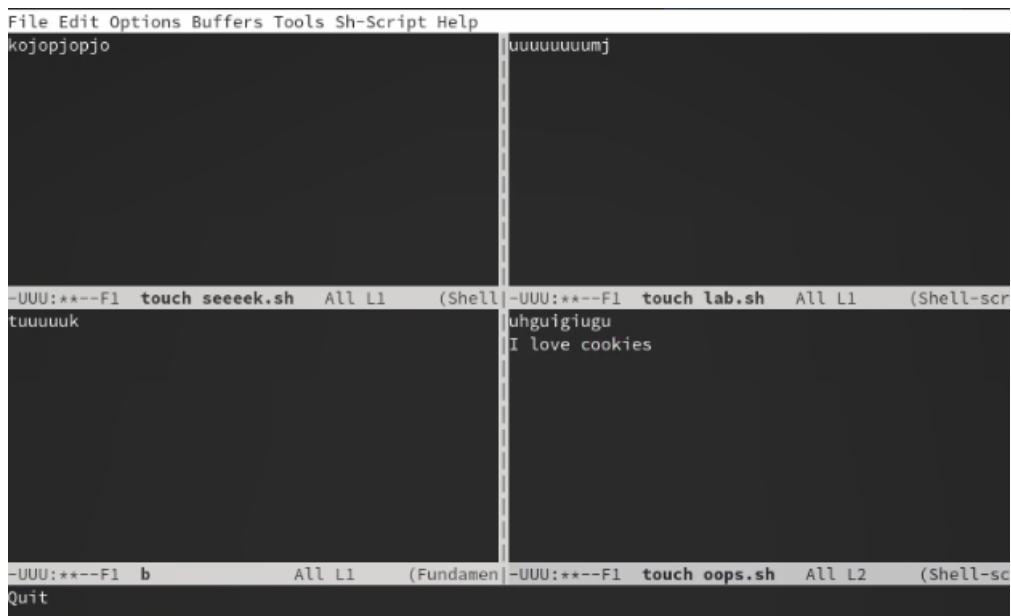


Рис. 22: Рис.22

9.4. Испробуйте другой режим поиска, нажав M-s o. Объясните, чем он отличается от обычного режима?

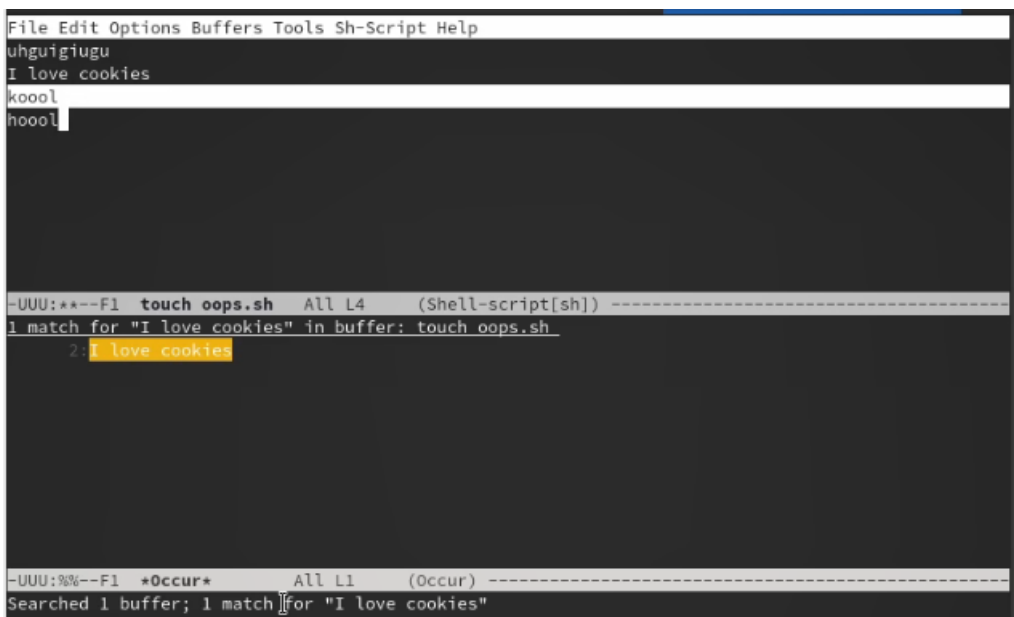


Рис. 23: Рис.23

## Вывод

Я познакомилась с операционной системой Linux и получила практические навыки работы с редактором Emacs.



# Контрольные вопросы

1. Кратко охарактеризуйте редактор emacs.

Emacs или EMACS (редактор макросов) это семейство текстовых редакторов, которые характеризуются своей расширяемостью. Руководство по наиболее широко используемому варианту, GNU Emacs, описывает его как “расширяемый, настраиваемый, самодокументируемый редактор отображения в реальном времени”.

2. Какие особенности данного редактора могут сделать его сложным для освоения новичком?

Emacs — действительно не сильно сложный редактор. Но это как раз тот пример, когда освоив минимум, нельзя на этом останавливаться, ибо сила emacs — в доступных библиотеках и дополнительных функциях. Единственный способ добиться максимального результата — это выучить комбинации клавиш, названия функций и сами по себе концепции и функционал.

3. Своими словами опишите, что такое буфер и окно в терминологии emacs’a.

Буфер (информатика) — область памяти, используемая для временного хранения данных ввода-вывода.

Окно - область экрана, в которой отображается буфер.

5. Какие буферы создаются по умолчанию при запуске emacs?

Scratch и Messages.

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

ctrl + c + shift + | и ctrl + c + ctrl + shift +

7. Как поделить текущее окно на две части?

ctrl + x + 2

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

Скажу честно, vi мне понравился больше как раз таки из-за удобства в использовании.