

Лабораторная работа №11

Операционные системы

Павлова Татьяна Юрьевна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	8
4	Выполнение лабораторной работы	9
5	Выводы	19

Список иллюстраций

4.1	Открытый emacs	9
4.2	lab07.sh	10
4.3	Сохранение файла	10
4.4	Вырезанная строка	11
4.5	Вставленная строка	11
4.6	Выделенная область текста	12
4.7	Вырезанная область текста	12
4.8	Отмена действий	13
4.9	Скопированная и вставленная в конец область	13
4.10	Курсор в начале строки	14
4.11	Курсор в конце строки	14
4.12	Список буферов	15
4.13	Переключение на другой буфер	15
4.14	Фрейм, поделенный на 4 части	16
4.15	Режим поиска	17
4.16	Переключение между результатами	18

Список таблиц

1 Цель работы

Целью данной лабораторной работы является ознакомление с операционной системой Linux, а также получение практические навыков работы с редактором Emacs.

2 Задание

1. Открыть emacs.
2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).
3. Наберите текст
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). 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-/).
6. Научитесь использовать команды по перемещению курсора. 6.1. Переместите курсор в начало строки (C-a). 6.2. Переместите курсор в конец строки (C-e). 6.3. Переместите курсор в начало буфера (M-<). 6.4. Переместите курсор в конец буфера (M->).
7. Управление буферами. 7.1. Вывести список активных буферов на экран (C-x C-b). 7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых буферов и переключитесь на другой буфер. 7.3. Закройте это окно (C-x 0). 7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b).
8. Управление окнами. 8.1. Поделите фрейм на 4 части: разделите фрейм на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по

горизонтали (С-х 2) 8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста.

9. Режим поиска 9.1. Переключитесь в режим поиска (С-s) и найдите несколько слов, присутствующих в тексте. 9.2. Переключайтесь между результатами поиска, нажимая С-s. 9.3. Выйдите из режима поиска, нажав С-g. 9.4. Перейдите в режим поиска и замены (М-%), введите текст, который следует найти и заменить, нажмите Enter , затем введите текст для замены. После того как будут подсвечены результаты поиска, нажмите ! для подтверждения замены. 9.5. Испробуйте другой режим поиска, нажав М-s o. Объясните, чем он отличается от обычного режима?

3 Теоретическое введение

Основные термины Emacs

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.

4 Выполнение лабораторной работы

Откройте emacs (рис. 1).

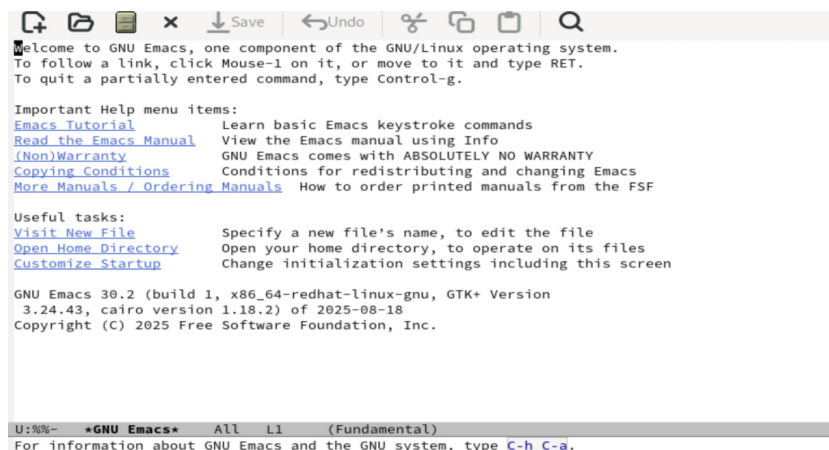
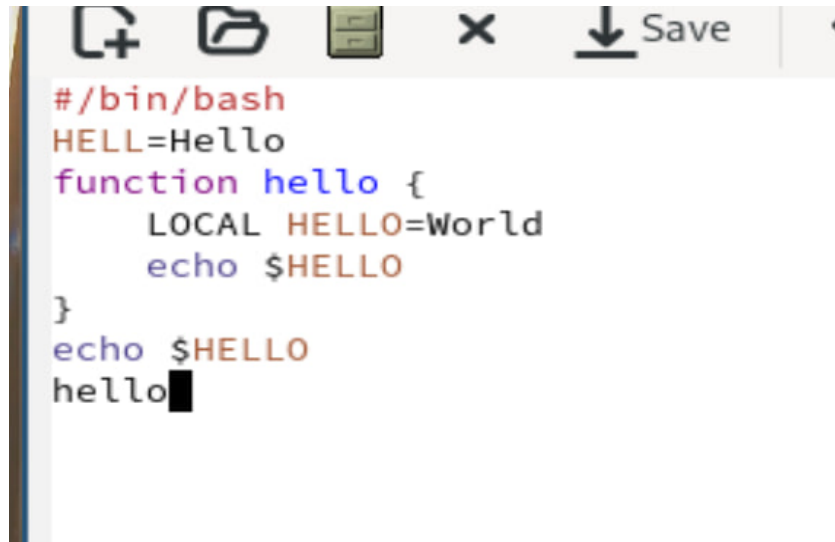


Рис. 4.1: Открытый emacs

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



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

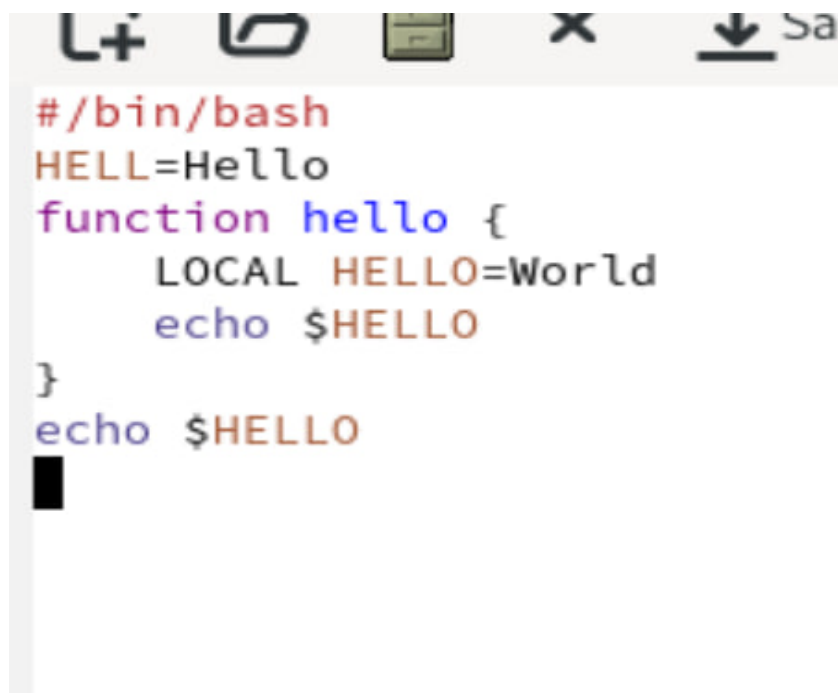
Рис. 4.2: lab07.sh

Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s) (рис. 3).



Рис. 4.3: Сохранение файла

Управление буферами. 7.1. Вывести список активных буферов на экран (C-x C-b). 7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых буферов и переключитесь на другой буфер. 7.3. Закройте это окно (C-x 0). 7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b). Прodelать с текстом стандартные процедуры редактирования, каждое действие должно 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-/). (рис. 4), (рис. 5), (рис. 6), (рис. 7), (рис. 8), (рис. 9).

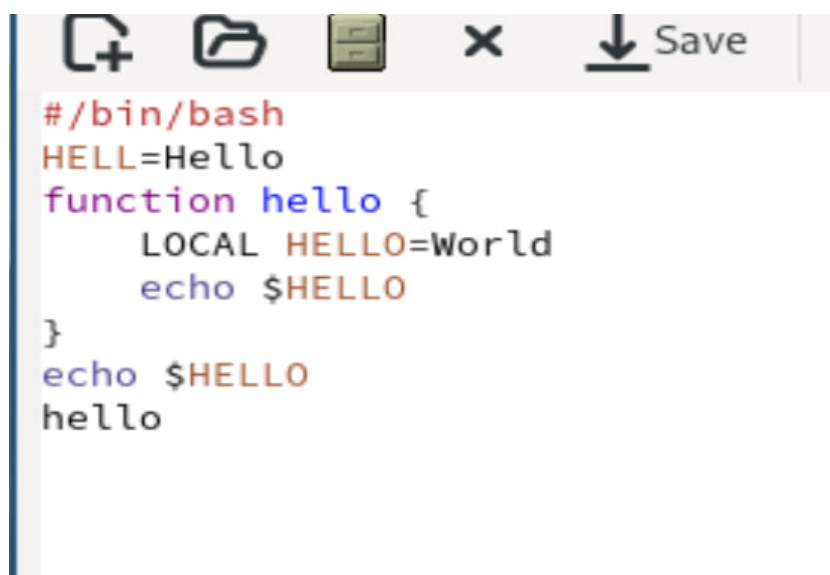


A screenshot of a code editor window. The title bar shows icons for copy, paste, save, close, and a 'Save' button. The code is as follows:

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

The cursor is at the end of the last line, indicated by a black block.

Рис. 4.4: Вырезанная строка



A screenshot of the same code editor window. The code is now as follows:

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

The word 'hello' has been added on a new line after 'echo \$HELLO'.

Рис. 4.5: Вставленная строка

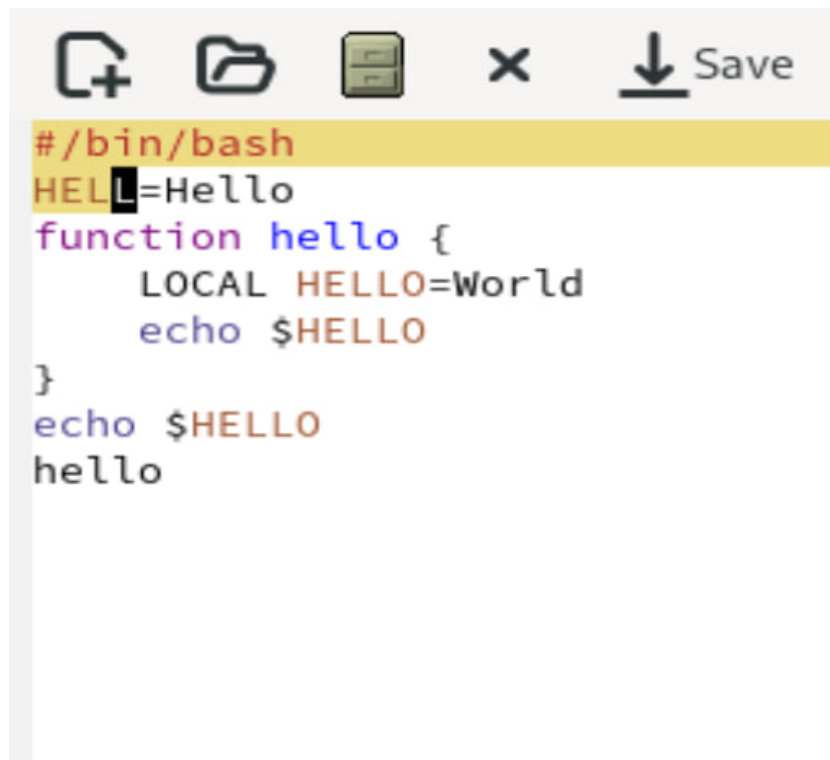


Рис. 4.6: Выделенная область текста

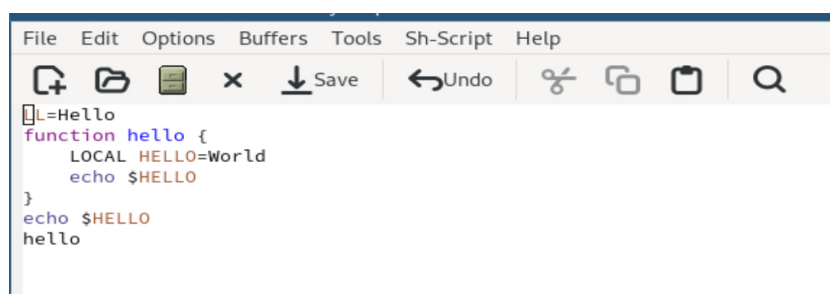


Рис. 4.7: Вырезанная область текста

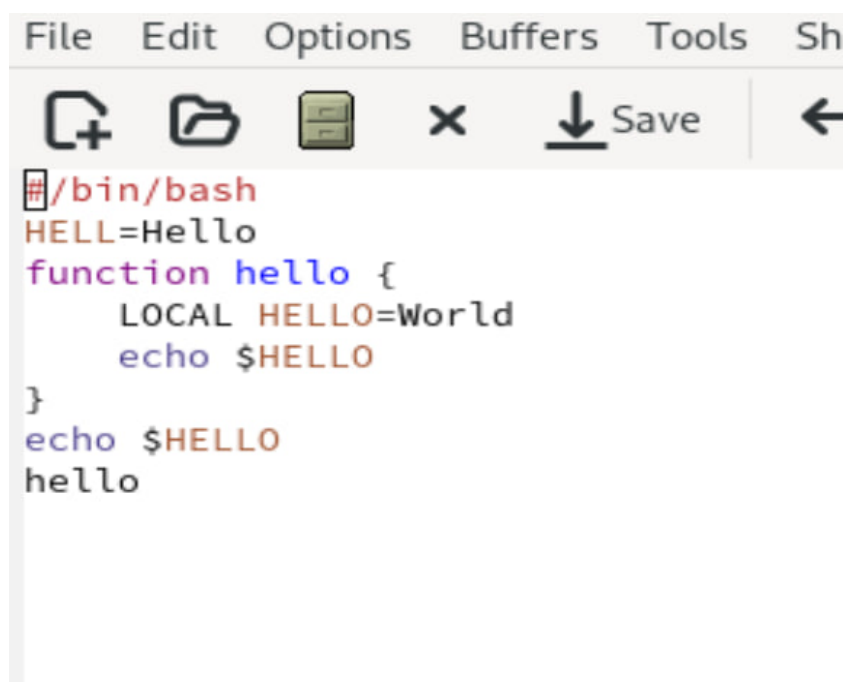


Рис. 4.8: Отмена действий

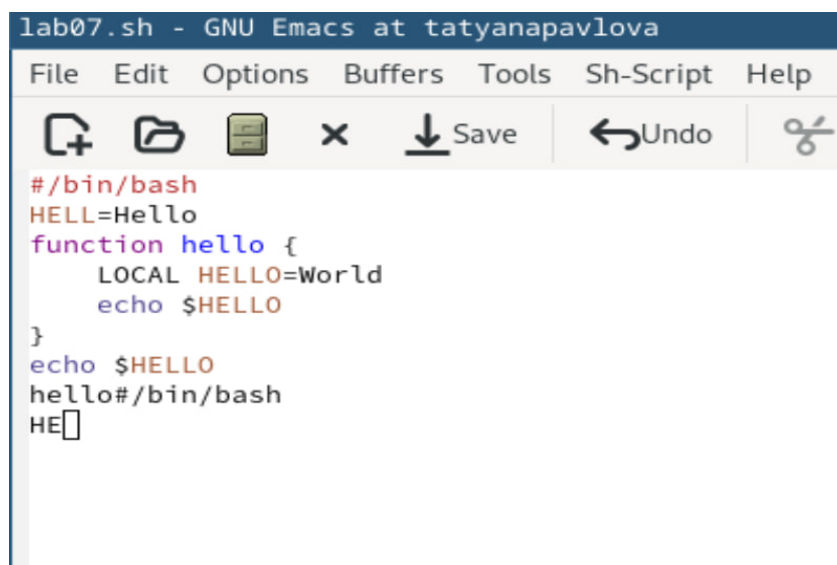


Рис. 4.9: Скопированная и вставленная в конец область

Научитесь использовать команды по перемещению курсора. 6.1. Переместите курсор в начало строки (C-a). 6.2. Переместите курсор в конец строки (C-e). 6.3. Переместите курсор в начало буфера (M-<). 6.4. Переместите курсор в конец буфера (M->) (рис. 10), (рис. 11).

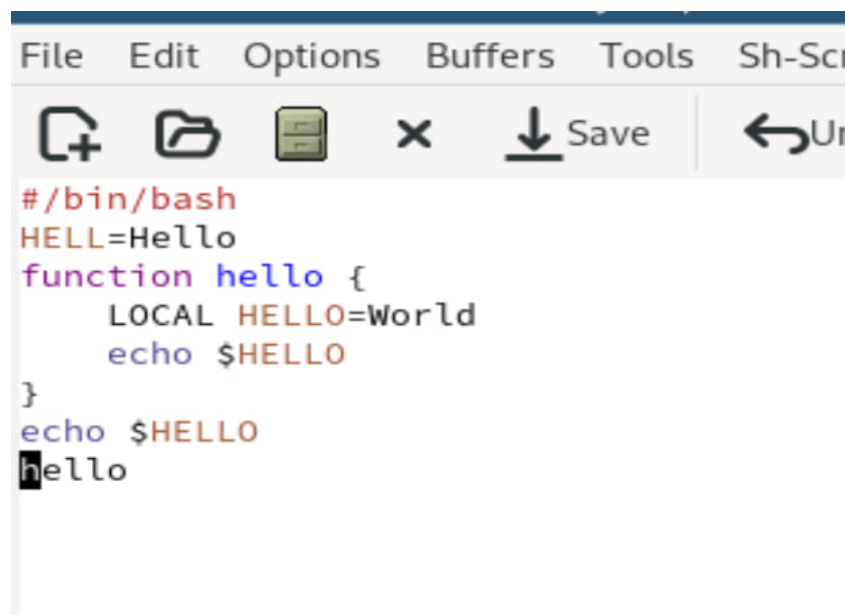


Рис. 4.10: Курсор в начале строки

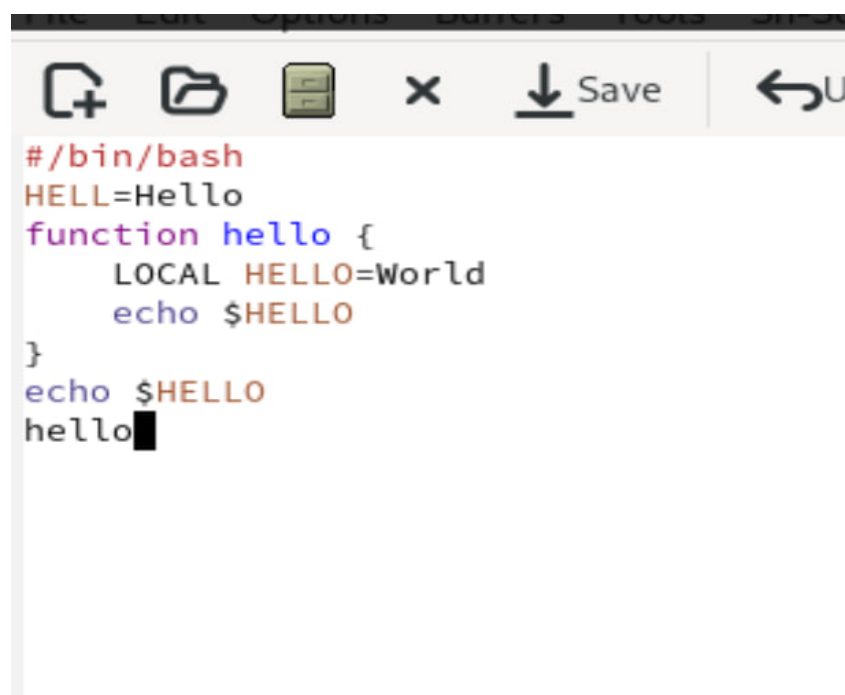


Рис. 4.11: Курсор в конце строки

Управление буферами. 7.1. Вывести список активных буферов на экран (C-x C-b). 7.2. Переместиться во вновь открытое окно (C-x) о со списком открытых

буферов и переключитесь на другой буфер. 7.3. Закройте это окно (C-x 0). 7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b) (рис. 12), (рис. 13).



Рис. 4.12: Список буферов

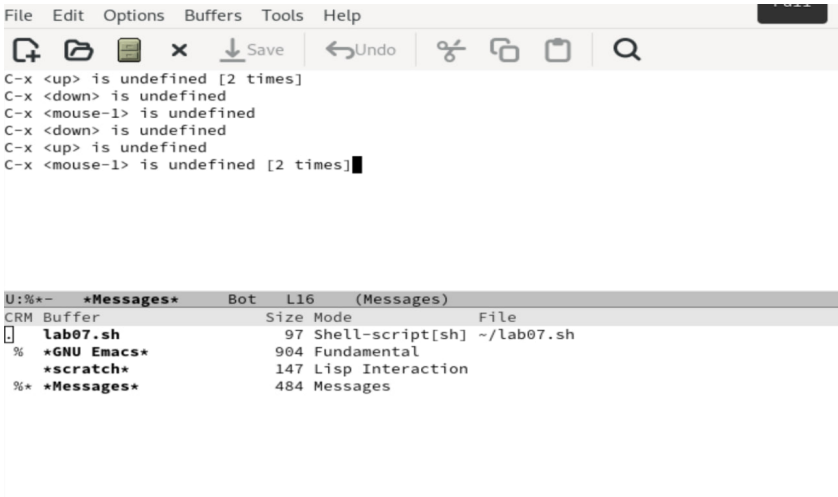


Рис. 4.13: Переключение на другой буфер

Управление окнами. 8.1. Поделите фрейм на 4 части: разделите фрейм на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по горизонтали (C-x 2) 8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста (рис. 14).

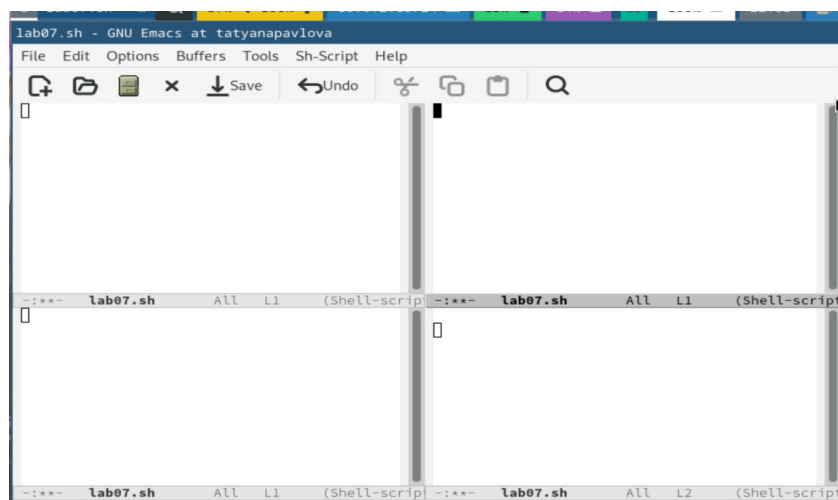


Рис. 4.14: Фрейм, поделенный на 4 части

Режим поиска 9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте. 9.2. Переключайтесь между результатами поиска, нажимая C-s. 9.3. Выйдите из режима поиска, нажав C-g. 9.4. Перейдите в режим поиска и замены (M-%), введите текст, который следует найти и заменить, нажмите Enter, затем введите текст для замены. После того как будут подсвечены результаты поиска, нажмите ! для подтверждения замены. 9.5. Испробуйте другой режим поиска, нажав M-s o (рис. 15), (рис. 16).

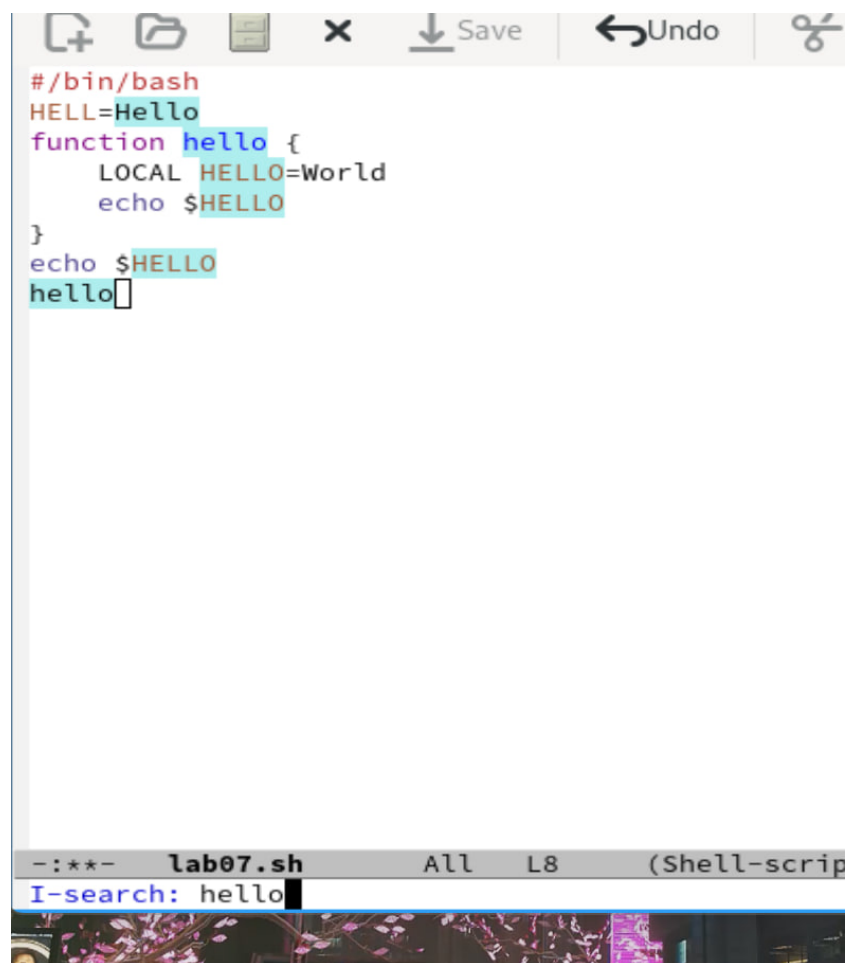
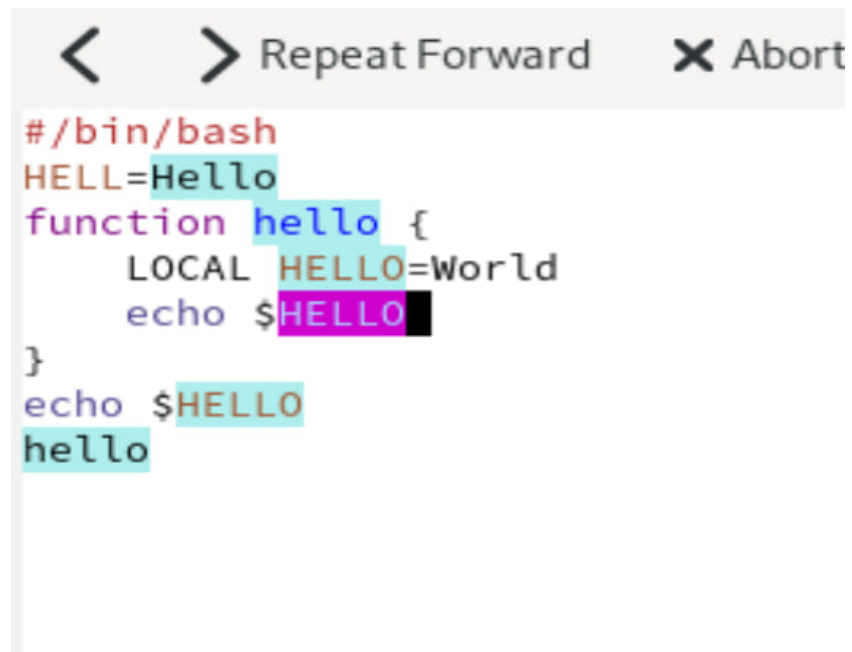


Рис. 4.15: Режим поиска



```
< > Repeat Forward X Abort
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELL
hello
```

Рис. 4.16: Переключение между результатами

5 Выводы

При выполнении данной лабораторной работы, я ознакомилась с операционной системой Linux, а также получила практические навыки работы с редактором Emacs.