

# **Лабораторная работа №9**

**Операционные системы**

Газизова Регина

# Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	21
4	Контрольные вопросы	22

## Список иллюстраций

2.1	Emacs . . . . .	6
2.2	Создание файла . . . . .	6
2.3	Текст . . . . .	7
2.4	Редактирование текста . . . . .	7
2.5	Вставка . . . . .	8
2.6	Выделение . . . . .	8
2.7	Вставка . . . . .	9
2.8	Удаление текста . . . . .	9
2.9	Отмена действия . . . . .	9
2.10	Курсор в начале строки . . . . .	10
2.11	Курсор в конец строки . . . . .	10
2.12	Начало буфера . . . . .	10
2.13	Конец буфера . . . . .	11
2.14	Список буферов . . . . .	11
2.15	Выбор буфера . . . . .	12
2.16	Перемещение . . . . .	12
2.17	Закрытие окна . . . . .	13
2.18	Перемещение . . . . .	14
2.19	На 2 части . . . . .	15
2.20	На 4 части . . . . .	15
2.21	Открытие новых файлов . . . . .	16
2.22	Режим поиска . . . . .	17
2.23	Переключение между результатами . . . . .	17
2.24	Выход из режима поиска . . . . .	18
2.25	Поиск и замена . . . . .	18
2.26	Замена . . . . .	19
2.27	Поиск 2.0 . . . . .	20

## **Список таблиц**

# 1 Цель работы

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

#Задание

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

## 2 Выполнение лабораторной работы

1. Открываем emacs через командную строку emacs & (рис.1)

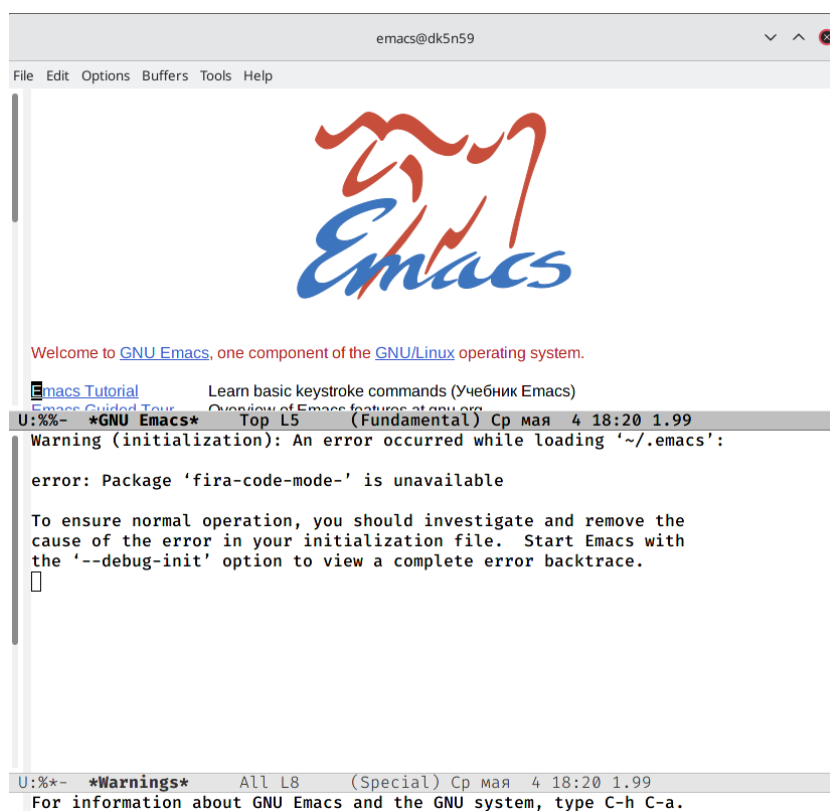


Рис. 2.1: Emacs

2. Создаем файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (рис.2)

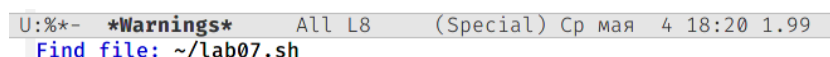
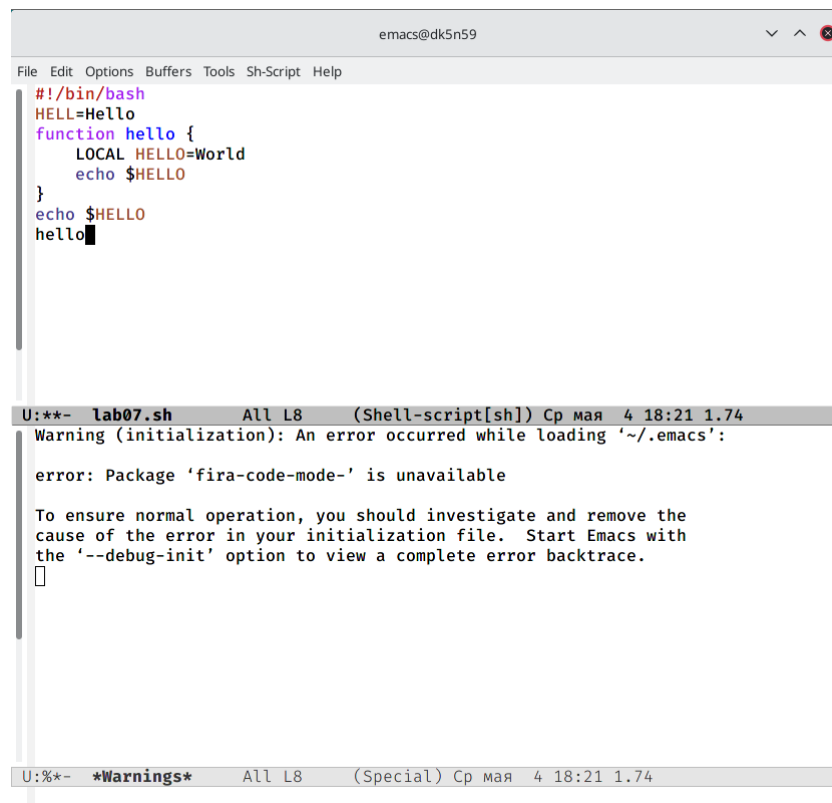


Рис. 2.2: Создание файла

3. Набираем предложенный текст (рис.3) и сохраняем файл с помощью комбинации Ctrl-x Ctrl-s



The screenshot shows the Emacs editor window titled 'emacs@dk5n59'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main text area contains a shell script:

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

Below the script, a status bar shows 'U:\*\*\*- lab07.sh All L8 (Shell-script[sh]) Ср мая 4 18:21 1.74'. A message box displays a warning and an error:

```
Warning (initialization): An error occurred while loading '~/.emacs':

error: Package 'fira-code-mode-' is unavailable

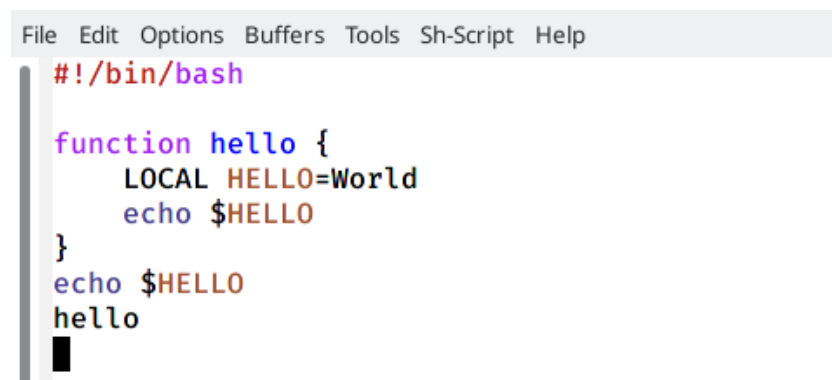
To ensure normal operation, you should investigate and remove the
cause of the error in your initialization file. Start Emacs with
the '--debug-init' option to view a complete error backtrace.
```

At the bottom, another status bar shows 'U:%\*- \*Warnings\* All L8 (Special) Ср мая 4 18:21 1.74'.

Рис. 2.3: Текст

4. Редактируем текст.

- 4.1. Вырежем одной командой строку с помощью комбинации Ctrl-k (рис.4)



This screenshot shows the Emacs editor window with the same shell script as in Figure 2.3. The menu bar is the same. The script content is:

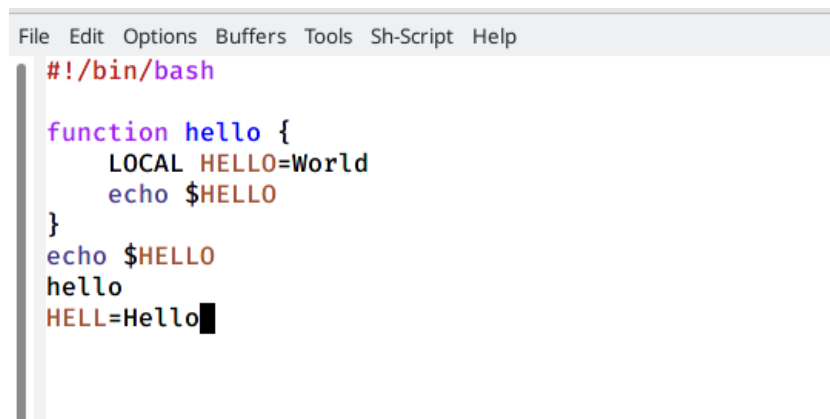
```
#!/bin/bash

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

The cursor is positioned at the end of the 'hello' line.

Рис. 2.4: Редактирование текста

4.2. Вставим эту строку в конец файла (Ctrl-y) (рис.5).



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

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

Рис. 2.5: Вставка

4.3. Выделим область текста (Ctrl-space) и скопируем область в буфер обмена (Alt-w) (рис.6).



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

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

Рис. 2.6: Выделение

4.4. Вставим область в конец файла (Ctrl-y) (рис.7).

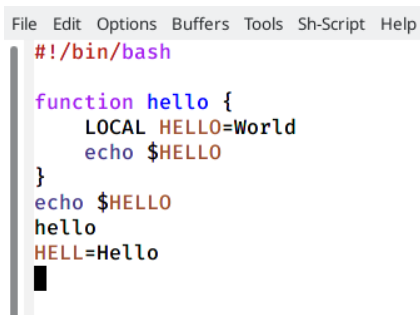


```
#!/bin/bash

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

Рис. 2.7: Вставка

4.5. Вновь выделим эту область и на этот раз вырежем её (Ctrl-w) (рис.8).

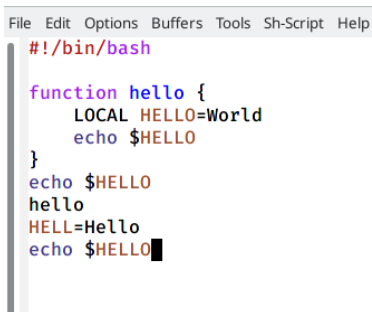


```
#!/bin/bash

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

Рис. 2.8: Удаление текста

4.6. Отменим последнее действие (Ctrl-/) (рис.9).



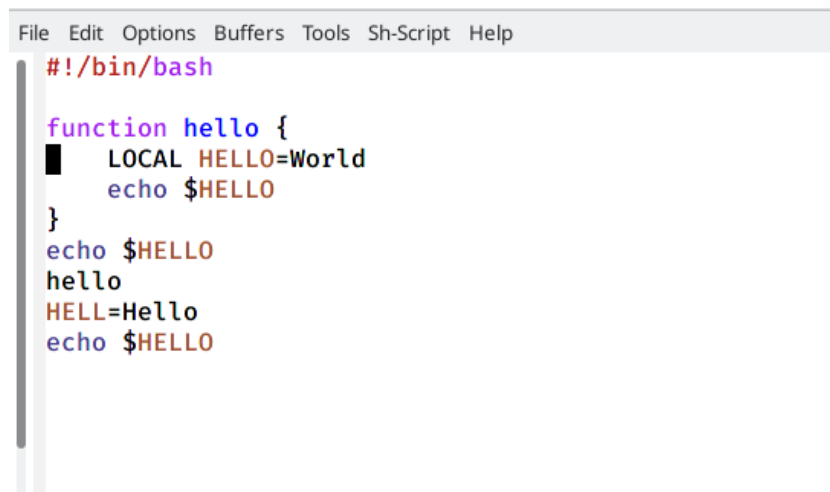
```
#!/bin/bash

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

Рис. 2.9: Отмена действия

5. Научимся пользоваться командами по перемещению курсора:

5.1. Переместим курсор в начало строки (Ctrl-a) (рис.10).

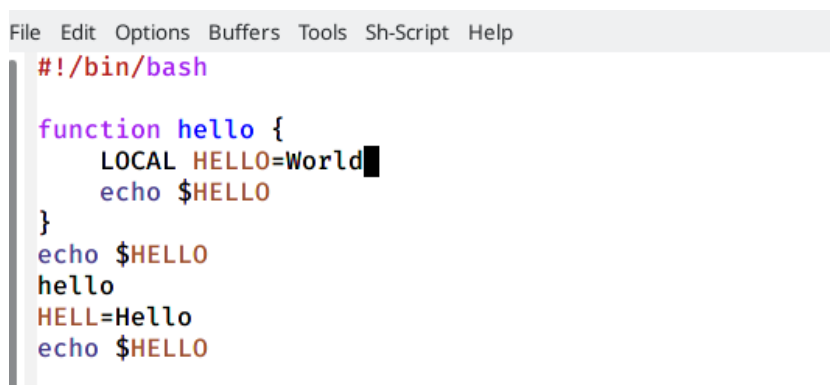


```
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash

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

Рис. 2.10: Курсор в начале строки

5.2. Переместим курсор в конец строки (Ctrl-e) (рис.11).

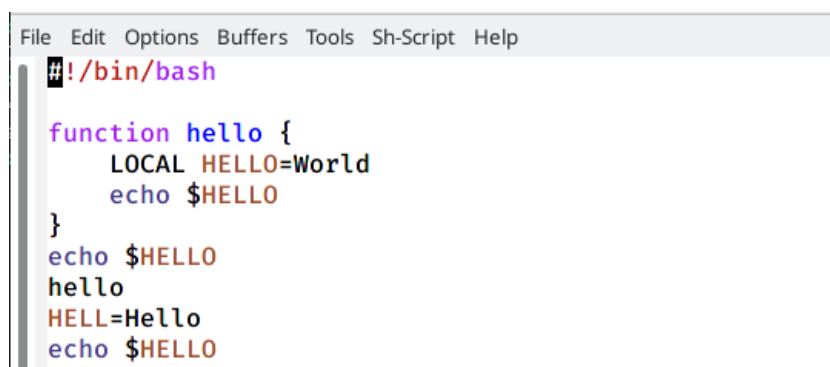


```
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash

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

Рис. 2.11: Курсор в конец строки

5.3. Переместим курсор в начало буфера (Alt-<) (рис.12).

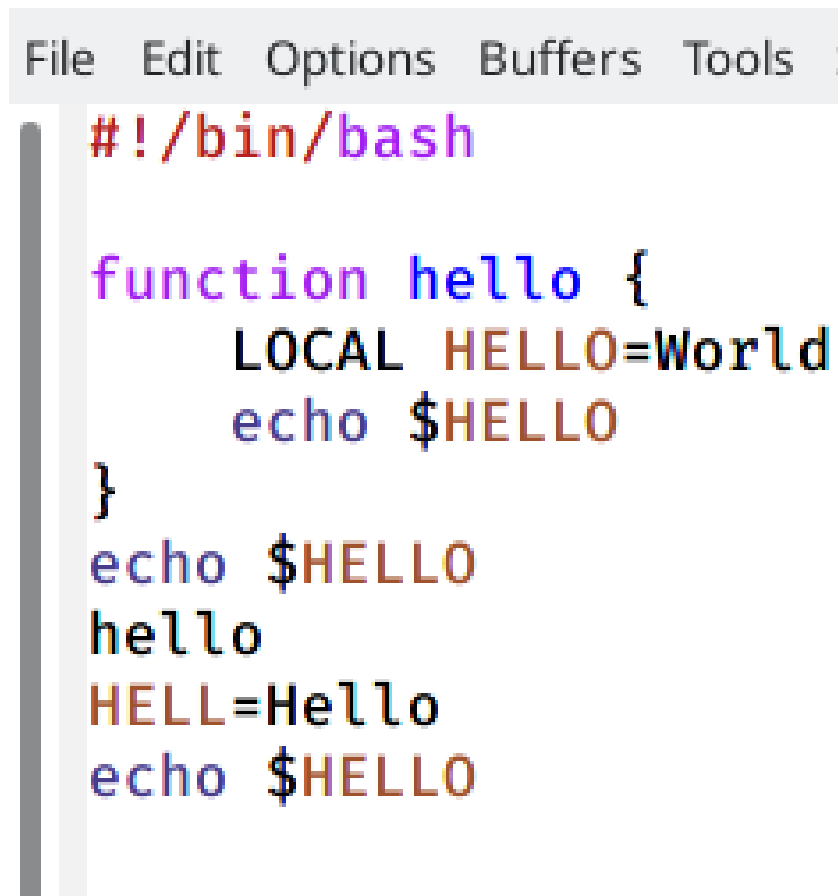


```
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash

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

Рис. 2.12: Начало буфера

5.4. Переместим курсор в конец буфера (Alt->) (рис.13).

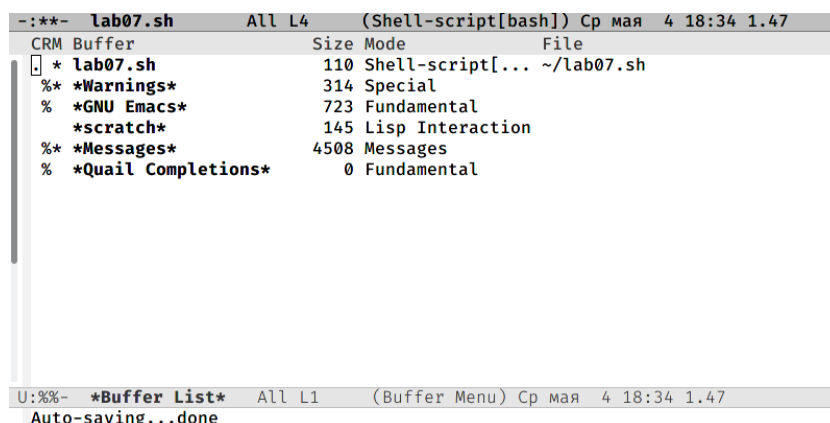


```
File Edit Options Buffers Tools :
#!/bin/bash

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

Рис. 2.13: Конец буфера

6. Выведем список активных буферов на экран (Ctrl-x Ctrl-b) (рис.14).



```
--:*** lab07.sh All L4 (Shell-script[bash]) Cp мая 4 18:34 1.47
CRM Buffer Size Mode File
[*] * lab07.sh 110 Shell-script[... ~/lab07.sh
%* *Warnings* 314 Special
% *GNU Emacs* 723 Fundamental
% *scratch* 145 Lisp Interaction
%* *Messages* 4508 Messages
% *Quail Completions* 0 Fundamental

U:%%- *Buffer List* All L1 (Buffer Menu) Cp мая 4 18:34 1.47
Auto-saving...done
```

Рис. 2.14: Список буферов

6.1. Переместимся во вновь открытое окно (Ctrl-x) со списком открытых буферов и переключимся на другой буфер (Ctrl-x b) (рис.15-16).

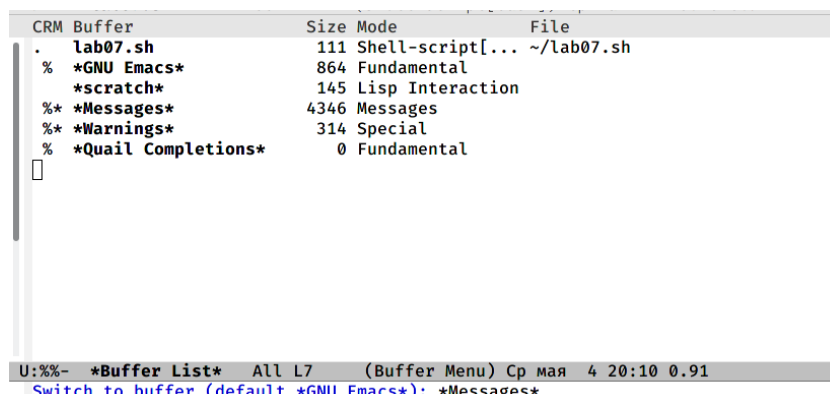


Рис. 2.15: Выбор буфера

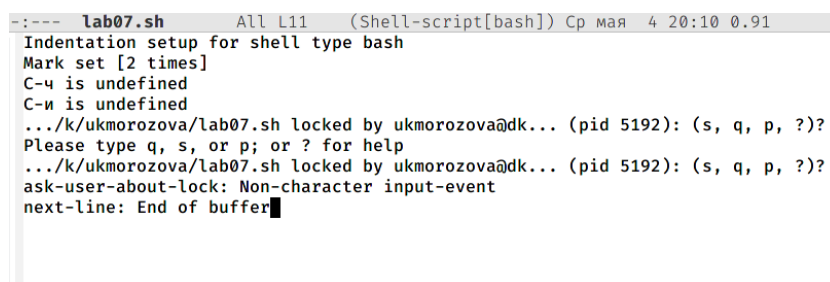


Рис. 2.16: Перемещение

6.2. Закроем это окно (Ctrl-x 0) (рис.17).



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

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO

-:--- lab07.sh All L11 (Shell-script[bash]) Ср мая
C-x 0
```

Рис. 2.17: Заккрытие окна

6.3. Теперь вновь переключимся между буферами, но уже без вывода их списка на экран (Ctrl-x b) (рис.18).

```
emacs@dk8n54
file Edit Options Buffers Tools Minibuf Help
#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO
█

-- lab07.sh All L11 (Shell-script[bash]) Ср мая 4 20:11 0.81
Switch to buffer (default *Messages*): *Messages█
```

Рис. 2.18: Перемещение

7. Поделим фрейм на 4 части: разделим сначала фрейм на два окна по вертикали (Ctrl-x 3) (рис.19), а затем каждое из этих окон на две части по горизонтали (Ctrl-x 2) (рис.20). В каждом из четырёх созданных окон откроем новый буфер (файл) и введем несколько строк текста (рис.21).

```
File Edit Options Buffers Tools Sh-Script Help

#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO

#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO

-:--- lab07.sh All L11 (Shell-
Loading /usr/share/site-lisp/rc.d/window-system/x/fontset.ecf...done
Loading /usr/share/site-lisp/rc.d/packages/font-lock.ecf...done
Loading /usr/share/site-lisp/rc.d/packages/font-lock-jit.ecf...done
Setting 'package-selected-packages' temporarily since "emacs -q" would overwrite
customizations
For information about GNU Emacs and the GNU system, type C-h C-a.
Setting up indent for shell type bash
Indentation variables are now local.
Indentation setup for shell type bash
C-x C-3 is undefined

U:%*- *Messages* Bot L59 (Messages) Ср мая 4 18:43 1.21
```

Рис. 2.19: На 2 части

```
File Edit Options Buffers Tools Sh-Script Help

echo $HELLO
hello
HELL=Hello
echo $HELLO

echo $HELLO
hello
HELL=Hello
echo $HELLO

 -:--- lab07.sh Bot L11 (Shell-
hello
HELL=Hello
echo $HELLO

 -:--- lab07.sh Bot L11 (Shell-s
hello
HELL=Hello
echo $HELLO

 -:--- lab07.sh Bot L11 (Shell-
 -:--- lab07.sh Bot L11 (Shell-s
```

Рис. 2.20: На 4 части

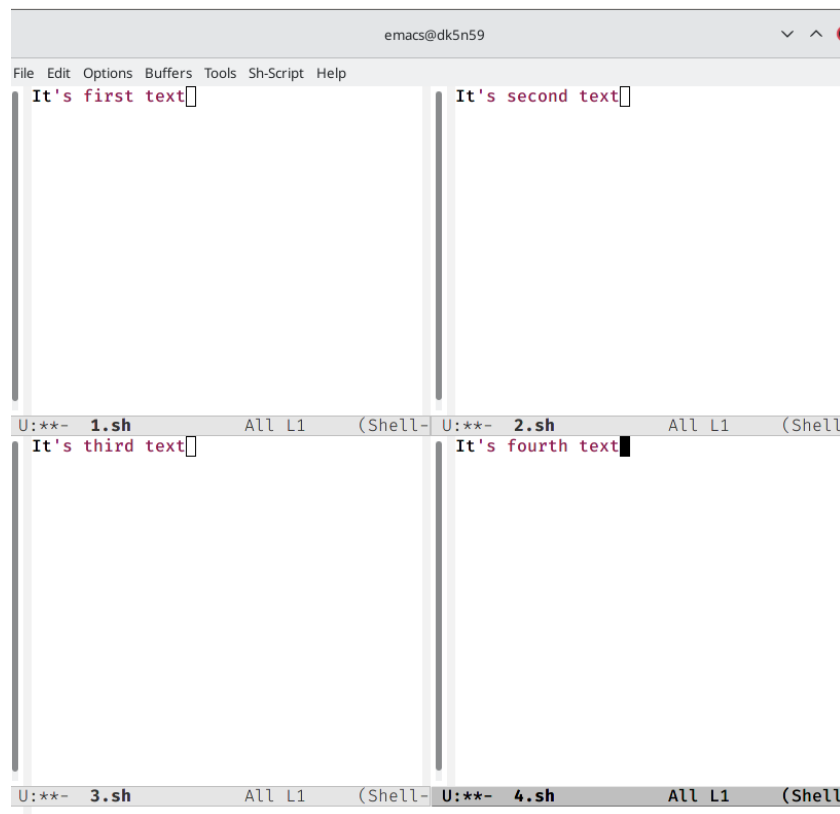


Рис. 2.21: Открытие новых файлов

8. Переключимся в режим поиска (Ctrl-s) и найдем несколько слов, присутствующих в тексте (рис.22).



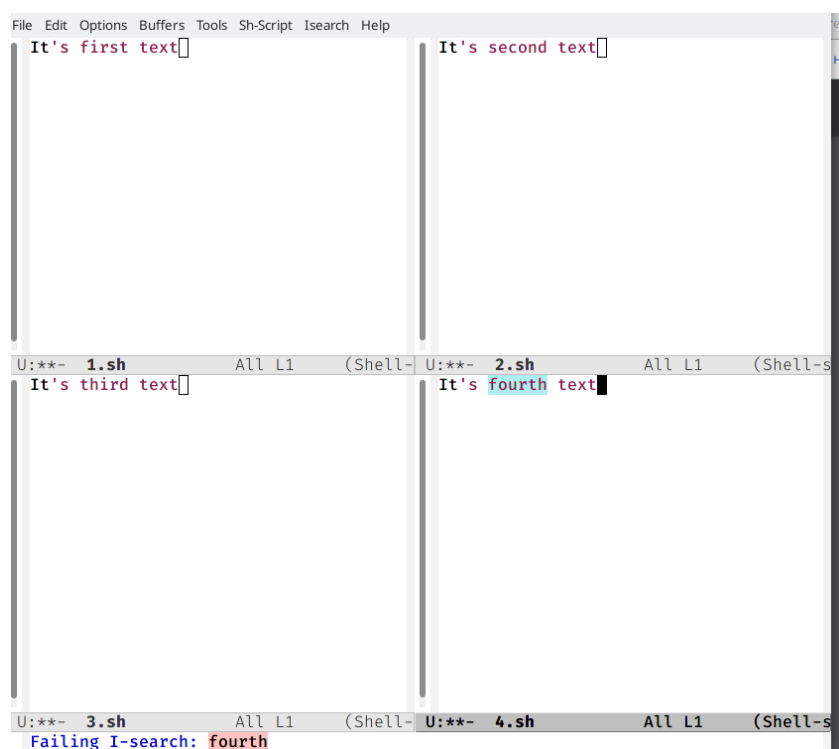


Рис. 2.22: Режим поиска

8.1. Переключимся между результатами поиска, нажимая Ctrl-s (рис.23).



Рис. 2.23: Переключение между результатами

8.2. Выйдем из режима поиска, нажав Ctrl-g (рис.24).

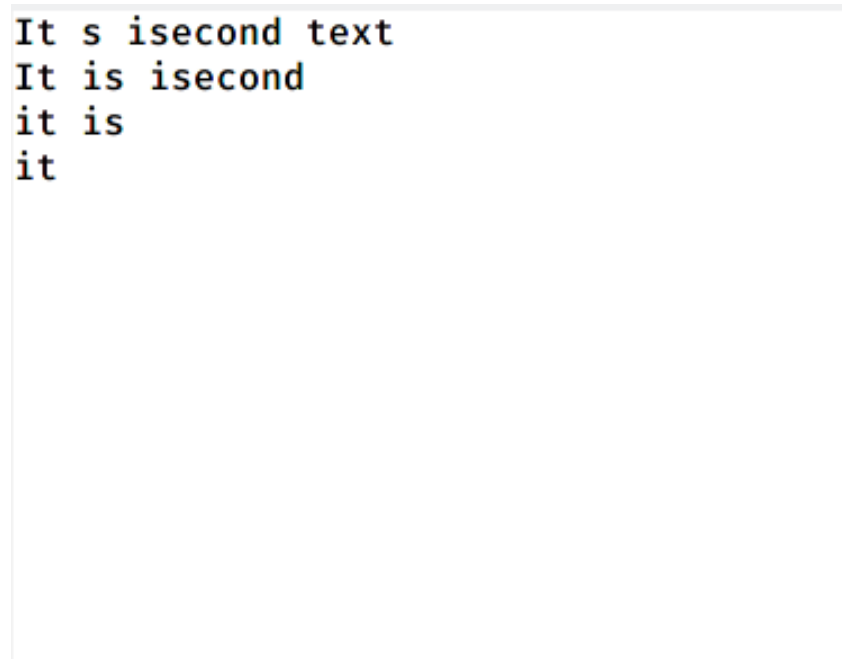


Рис. 2.24: Выход из режима поиска

8.3. Перейдем в режим поиска и замены (Alt-%), введем текст, который следует найти и заменить, нажмем Enter, затем введем текст для замены (рис.25). После того как будут подсвечены результаты поиска, нажмем ! для подтверждения замены (рис.26).



Рис. 2.25: Поиск и замена



```
It s issecond text
It is issecond
it is
it
```

Рис. 2.26: Замена

8.4. Испробуем другой режим поиска, нажав Alt-s о (рис.27).

The screenshot shows the Emacs editor interface with a file named `lab07.sh` open. The editor's menu bar includes `File`, `Edit`, `Options`, `Buffers`, `Tools`, `Minibuf`, and `Help`. The main window displays a shell script with the following content:

```
#!/bin/bash

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

Below the script, the output of running the script is shown in a separate buffer. The output includes messages about indentation setup, mark setting, and search failures:

```
-- lab07.sh All L11 (Shell-script[bash]) Ср мая 4 20:19 0.92
Indentation setup for shell type bash
Mark set [2 times]
C-ч is undefined
C-н is undefined
.../k/ukmorofova/lab07.sh locked by ukmorofova@dk... (pid 5192): (s, q, p, ?)
Please type q, s, or p; or ? for help
.../k/ukmorofova/lab07.sh locked by ukmorofova@dk... (pid 5192): (s, q, p, ?)
ask-user-about-lock: Non-character input-event
next-line: End of buffer
C-c is undefined
C-c b is undefined
funcall-interactively: Search failed: "o"
funcall-interactively: Search failed: "hello"
```

At the bottom of the Emacs window, the status bar shows the current buffer is `*Messages*`, the cursor is at line 70, and the file is `(Messages)`. The date and time are `Ср мая 4 20:19 0.92`. A `Search:` prompt is visible at the very bottom.

Рис. 2.27: Поиск 2.0

## **3 Выводы**

Я приобрела практические навыки в работе с редактором etacs.

## 4 Контрольные вопросы

1). Emacs – один из наиболее мощных и широко распространённых редакторов, используемых в мире Unix. По популярности он соперничает с редактором vi и его клонами. В зависимости от ситуации, Emacs может быть текстовым редактором; программой для чтения почты и новостей Usenet; интегрированной средой разработки (IDE); операционной системой и т.д. Всё это разнообразие достигается благодаря архитектуре Emacs, которая позволяет расширять возможности редактора при помощи языка Emacs Lisp. На языке C написаны лишь самые базовые и низкоуровневые части Emacs, включая полнофункциональный интерпретатор языка Lisp. Таким образом, Emacs имеет встроенный язык программирования, который может использоваться для настройки, расширения и изменения поведения редактора. В действительности, большая часть того редактора, с которым пользователи Emacs работают в наши дни, написана на языке Lisp.

2). Основную трудность для новичков при освоении данного редактора могут составлять большое количество команд, комбинаций клавиш, которые не получится все запомнить с первого раза и поэтому придется часто обращаться к справочным материалам.

3). Буфер – это объект, представляющий собой текст. Если имеется несколько буферов, то редактировать можно только один. Обычно буфер считывает данные из файла или записывает в файл данные из буфера. Окно – это область экрана, отображающая буфер. При запуске редактора отображается одно окно, но при обращении к некоторым функциям могут открыться дополнительные окна. Окна Emacs и окна графической среды XWindow – разные вещи. Одно окно XWindow-

может быть разбито на несколько окон в смысле Emacs, в каждом из которых отображается отдельный буфер.


4). Да, можно.

5). При запуске Emacs по умолчанию создаются следующие буферы: «scratch» (буфер для несохраненного текста) «Messages» (журнал ошибок, включающий также информацию, которая появляется в области EchoArea) «GNU Emacs» (справочный буфер о редакторе).

6). С-с | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и нажимаю «|» С-сC-| сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и, удерживая «ctrl», нажимаю «|».

7). Чтобы разделить окно на две части необходимо воспользоваться комбинацией «Ctrl-x 3» (по вертикали) или «Ctrl-x 2» (по горизонтали).

8). Настройки Emacs хранятся в файле .emacs.

9). По умолчанию клавиша «» удаляет символ перед курсором, но в редакторе её можно переназначить. Для этого необходимо изменить конфигурацию файла .emacs.

10). Более удобным я считаю редактор emacs, потому что в нем проще открывать другие файлы, можно использовать сразу несколько окон, нет «Командного режима», «Режима ввода», «Режима командной строки», которые являются немного непривычными и в какой-то степени неудобным.