

Лабораторная работа №11

Барбакова А. С. -

Российский университет дружбы народов, Москва, Россия

Информация

- Барбакова Алиса Саяновна
- НКАбд-01-24, студ. билет - 1132246727
- Российский университет дружбы народов
- https://github.com/ASBarbakova/study_2024-2025_os-intro



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

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

Emacs представляет собой мощный экраный редактор текста, написанный на языке высокого уровня Elisp.

Выполнение лабораторной работы

Выполнение лабораторной работы

1. Устанавливаю emacs.

```
info x86_64 7.1.1-1.fc41 updates 361.7 KiB
libtree-sitter-java x86_64 0.21.0-2.fc41 fedora 465.3 KiB

Сводка транзакции:
Установка: 12 пакетов

Общий размер входящих пакетов составляет 101 MiB. Необходимо загрузить 101 MiB.
После этой операции будут использоваться дополнительные 362 MiB (установка 362 MiB, удаление 0 B).
Is this ok [y/N]: y
[ 1/12] gdk-pixbuf2-modules-extra-0:2.4 100% | 258.3 KiB/s | 86.8 KiB | 00m00s
[ 2/12] libotf-0:0.9.16-5.fc41.x86_64 100% | 256.6 KiB/s | 103.4 KiB | 00m00s
[ 3/12] libtree-sitter-0:0.23.0-2.fc41. 100% | 642.6 KiB/s | 108.0 KiB | 00m00s
[ 4/12] libXaw-0:1.0.16-2.fc41.x86_64 100% | 779.3 KiB/s | 202.6 KiB | 00m00s
[ 5/12] emacsclient-1:29.4-44.fc41.x86 100% | 346.2 KiB/s | 41.5 KiB | 00m00s
```

Рис. 1: Установка emacs

Выполнение лабораторной работы

Открываю emacs.

```
*GNU Emacs* - GNU Emacs at vbox

foot

(emacs:2875): Gtk-CRITICAL **: 21:57:46.416: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:47.011: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:47.579: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:48.249: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:48.279: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:48.291: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:48.315: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:48.344: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:48.357: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

(emacs:2875): Gtk-CRITICAL **: 21:57:48.368: gtk_distributional_allocation: assertion 'extra_space >= 0' failed

[asbarbakova@vbox ~]$ emacs
```

GNU Emacs - GNU Emacs at vbox

File Edit Options Buffers Tools Help

Welcome to GNU Emacs, one component of the GNU/Linux operating system.

To follow a link, click Mouse-1 on it, or move to it and type RET.

To quit a partially entered command, type Control-g.

Important Help menu items:

- [Emacs Tutorial](#) Learn basic Emacs keystroke commands
- [Read the Emacs Manual](#) View the Emacs manual using Info
- [\(Non\)Warranty](#) GNU Emacs comes with ABSOLUTELY NO WARRANTY
- [Copying Conditions](#) Conditions for redistribution and changing Emacs
- [More Manuals / Ordering Manuals](#) How to order printed manuals from the FSF

Useful tasks:

- [Visit New File](#) Specify a new file's name, to edit the file
- [Open Home Directory](#) Open your home directory, to operate on its files
- [Customize Startup](#) Change initialization settings including this screen

2. Создаю файл lab11.sh.

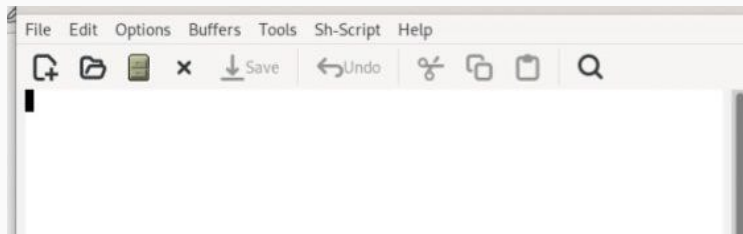
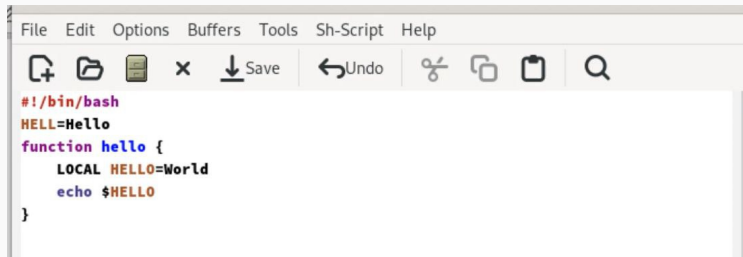


Рис. 3: Emacs

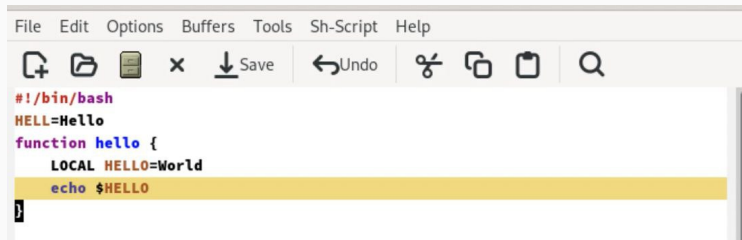
3. Набираю нужный текст.

A screenshot of the Emacs text editor interface. The menu bar at the top includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. Below the menu is a toolbar with icons for opening a file, saving, closing, undo, redo, copy, paste, and search. The main text area contains a shell script with the following content:

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

Рис. 4: Emacs

4. Сохраняю файл.
5. Проделываю с текстом стандартные процедуры редактирования, осуществляя каждое действие комбинацией клавиш:
 - 5.1 Вырезать одной командой целую строку (C-k).
 - 5.2 Вставить эту строку в конец файла (C-y).
 - 5.3 Скопировать область в буфер обмена (M-w).
 - 5.4 Вставить область в конец файла.
 - 5.5 Вновь выделить эту область и на этот раз вырезать её (C-w).
 - 5.6 Отмените последнее действие (C-/).

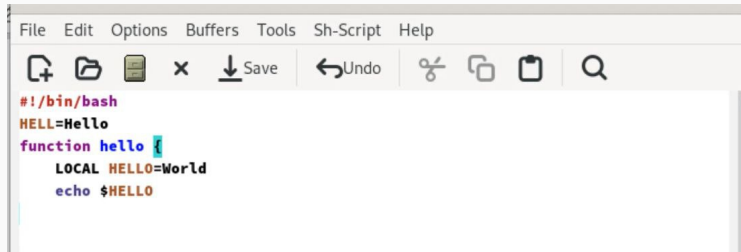


The image shows a screenshot of a text editor window titled "Sh-Script". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". The toolbar contains icons for file operations (new, open, save, close) and editing (undo, redo, copy, paste, search). The script content is as follows:

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

The line `echo $HELLO` is highlighted in yellow. A cursor is positioned at the end of the last line of the script.

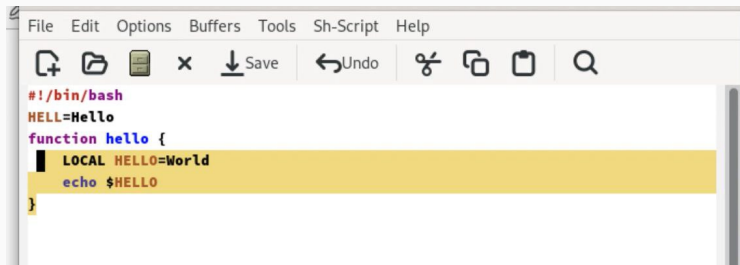
Рис. 5: Шаг 1



The image shows a screenshot of a Sh-Script editor window. The window has a menu bar with 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. Below the menu bar is a toolbar with icons for opening, saving, undo, redo, and search, along with text labels 'Save', 'Undo', and a magnifying glass icon. The main text area contains the following shell script code:

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

Рис. 6: Шаг 2

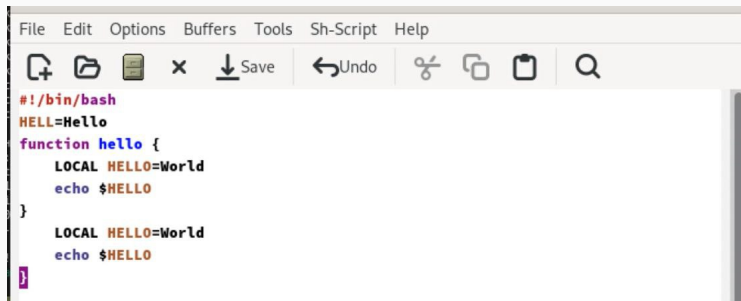


The image shows a screenshot of a Sh-Script editor window. The window has a menu bar with 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. Below the menu bar is a toolbar with icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search. The main text area contains a bash script with the following content:

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

The line 'LOCAL HELLO=World' is highlighted in yellow.

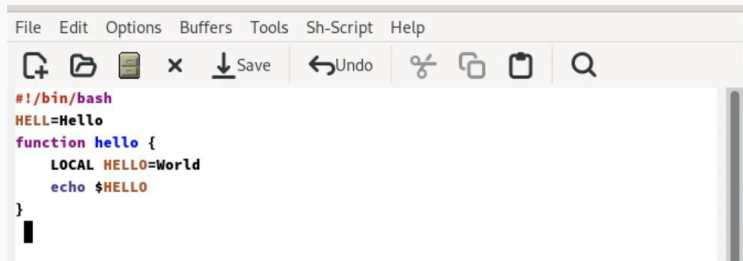
Рис. 7: Шаг 3



The image shows a screenshot of a text editor window. The title bar contains the text "File Edit Options Buffers Tools Sh-Script Help". Below the title bar is a toolbar with icons for file operations (new, open, save, close) and editing (undo, redo, copy, paste, search). The main text area contains the following shell script code:

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

Рис. 8: Шаг 4

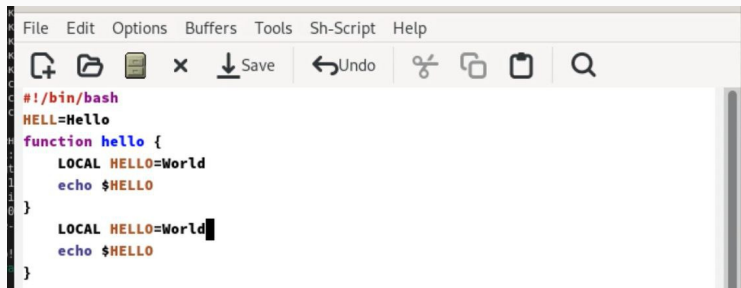


The image shows a screenshot of a text editor window. The title bar at the top contains the menu items: File, Edit, Options, Buffers, Tools, Sh-Script, and Help. Below the title bar is a toolbar with icons for opening a file, saving a file, closing a file, saving, undo, cut, copy, paste, and search. The main text area contains the following code:

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

The cursor is positioned at the end of the last line of code.

Рис. 9: Шаг 5



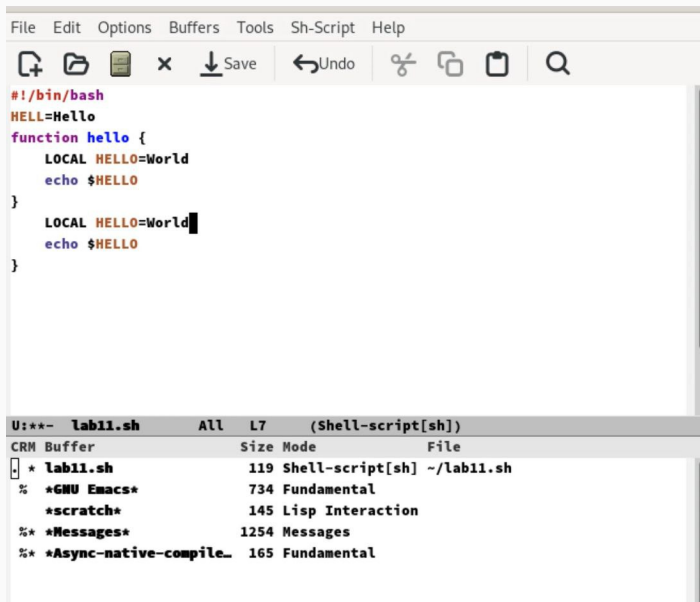
The image shows a screenshot of a text editor window. The title bar at the top reads "File Edit Options Buffers Tools Sh-Script Help". Below the title bar is a toolbar with icons for opening, saving, undo, redo, and search, along with text labels "Save" and "Undo". The main text area contains a shell script with the following content:

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

The script defines a function named "hello" that sets a local variable "HELLO" to "World" and prints it. After the function definition, the variable "HELLO" is set to "World" again and printed. The cursor is positioned at the end of the second "echo" command.

Рис. 10: Шаг 6

Выполнение лабораторной работы



The screenshot shows the Emacs editor interface. The menu bar includes File, Edit, Options, Buffers, Tools, Sh-Script, and Help. The toolbar contains icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search. The main text area displays a shell script with the following content:

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

Below the text area is a status bar showing the current buffer and window information: U:*** lab11.sh All L7 (Shell-script[sh]). Below the status bar is a list of buffers:

CRM	Buffer	Size	Mode	File
1	* lab11.sh	119	Shell-script[sh]	~/lab11.sh
%	*GNU Emacs*	734	Fundamental	
%	*scratch*	145	Lisp Interaction	
%*	*Messages*	1254	Messages	
%*	*Async-native-compile...	165	Fundamental	

6. Учиться пользоваться командами по перемещению курсора.

7. Управление буферами:

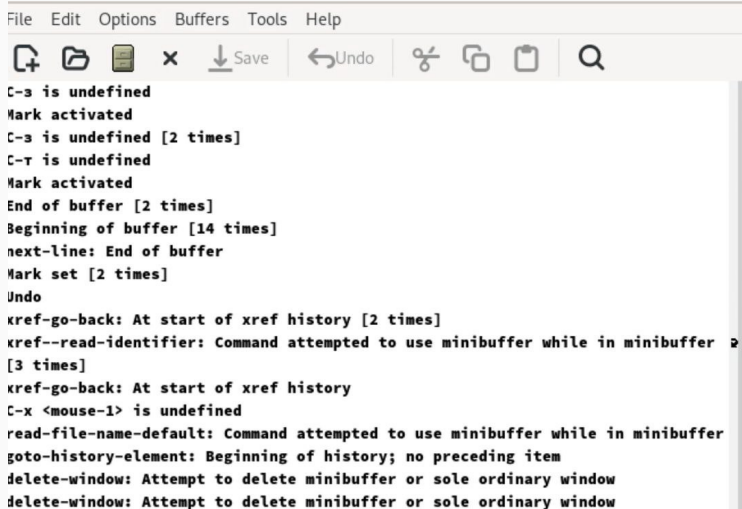
7.1 Вывести список активных буферов на экран (C-x C-b).

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).

Выполнение лабораторной работы



The screenshot shows the Emacs editor interface. The menu bar at the top includes File, Edit, Options, Buffers, Tools, and Help. The toolbar contains icons for saving, opening, closing, undo, redo, and search. The main text area displays a list of Emacs commands and their frequencies, such as 'C-3 is undefined', 'Mark activated', 'C-3 is undefined [2 times]', 'C-r is undefined', 'Mark activated', 'End of buffer [2 times]', 'Beginning of buffer [14 times]', 'next-line: End of buffer', 'Mark set [2 times]', 'Undo', 'xref-go-back: At start of xref history [2 times]', 'xref--read-identifier: Command attempted to use minibuffer while in minibuffer 2 [3 times]', 'xref-go-back: At start of xref history', 'C-x <mouse-1> is undefined', 'read-file-name-default: Command attempted to use minibuffer while in minibuffer', 'goto-history-element: Beginning of history; no preceding item', and 'delete-window: Attempt to delete minibuffer or sole ordinary window'.

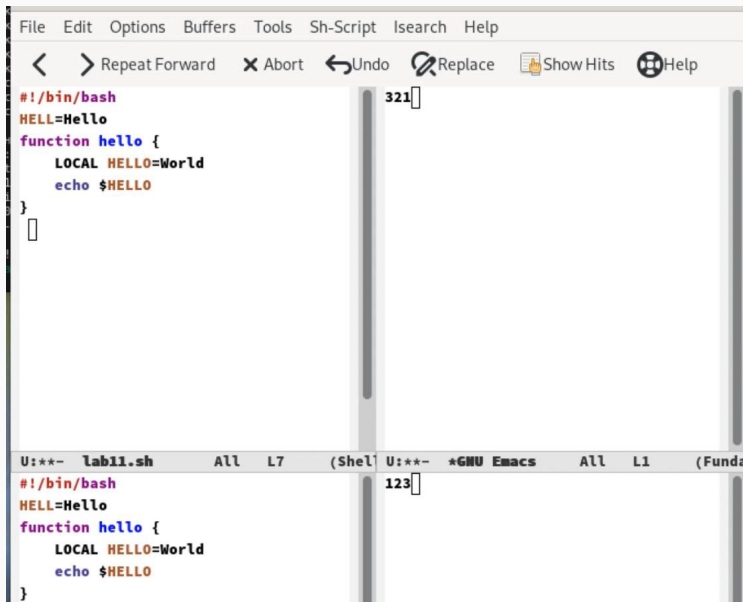
```
File Edit Options Buffers Tools Help
[Icons: Save, Open, Close, Undo, Redo, Search]

C-3 is undefined
Mark activated
C-3 is undefined [2 times]
C-r is undefined
Mark activated
End of buffer [2 times]
Beginning of buffer [14 times]
next-line: End of buffer
Mark set [2 times]
Undo
xref-go-back: At start of xref history [2 times]
xref--read-identifier: Command attempted to use minibuffer while in minibuffer 2
[3 times]
xref-go-back: At start of xref history
C-x <mouse-1> is undefined
read-file-name-default: Command attempted to use minibuffer while in minibuffer
goto-history-element: Beginning of history; no preceding item
delete-window: Attempt to delete minibuffer or sole ordinary window
delete-window: Attempt to delete minibuffer or sole ordinary window
```

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

Выполнение лабораторной работы



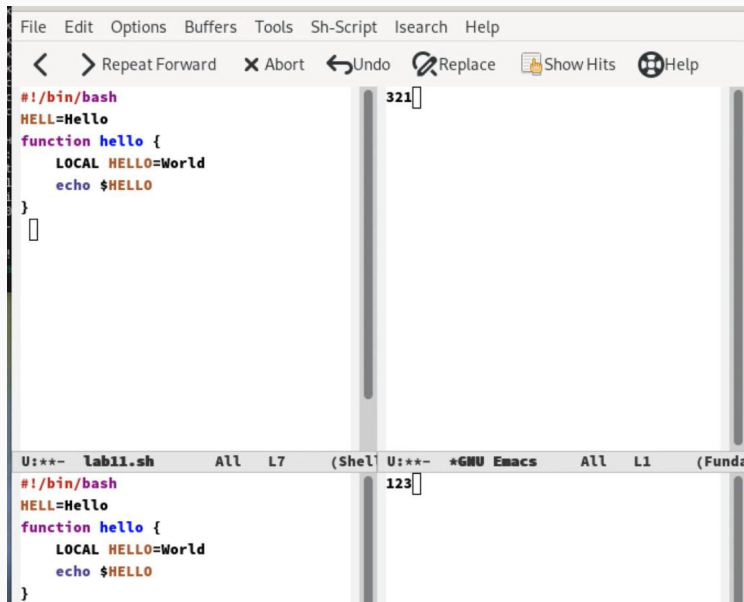
The image shows a GNU Emacs editor window with two buffers. The left buffer, named 'lab11.sh', contains a shell script. The right buffer, named '*GNU Emacs', is currently empty. The status bar at the bottom indicates the current buffer and line information.

```
File Edit Options Buffers Tools Sh-Script Isearch Help
< > Repeat Forward X Abort ↶ Undo ↷ Replace 📄 Show Hits 🛠 Help

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

U:*** lab11.sh All L7 (Shell) U:*** *GNU Emacs All L1 (Fundam
#!/bin/bash
HELL=Hello
function hello {
  LOCAL HELLO=World
  echo $HELLO
}
```

Выполнение лабораторной работы



The image shows a GNU Emacs editor window with a menu bar (File, Edit, Options, Buffers, Tools, Sh-Script, Isearch, Help) and a toolbar with icons for navigation and editing. The main text area contains a shell script with syntax highlighting. The script defines a function named 'hello' that prints 'Hello World'. The cursor is at the end of the script. The status bar at the bottom shows the file name 'lab11.sh' and the buffer name '*GNU Emacs'. A separate window or buffer is visible below, showing the output of the script: '321' and '123'.

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

321
```

U:*** lab11.sh All L7 (Shell) U:*** *GNU Emacs All L1 (Fundamental)

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

123
```


9. 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. Объясните, чем он отличается от обычного режима? Ответ - это режим поиска построчно.

```
U:***- lab11.sh All L7 (Shell)
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

U:***- lab11.sh All L7 (Shell)
Failing I-search: e
```

Выводы

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

Список литературы

1. Кулябов Д. С. Введение в операционную систему UNIX - Лекция.
2. Таненбаум Э., Бос Х. Современные операционные системы. - 4-е изд. -СПб. : Питер, 2015. - 1120 с.
3. Архитектура ЭВМ