

Лабораторная работа №11

Дисциплина - Операционные Системы

Азарцова В. В.

26 апреля 2025

Российский университет дружбы народов, Москва, Россия

Преподаватель Кулябов Д. С.

Информация

- Азарцова Вероника Валерьевна
- НКАбд-01-24, студ. билет №1132246751
- Российский университет дружбы народов
- 1132246751@pfur.ru
- <https://github.com/vvazarcova>

Целью данной лабораторной работы было познакомиться с операционной системой Linux. Получить практические навыки работы с редактором Emacs.

Задачи лабораторной работы:

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

Emacs — один из наиболее мощных и широко распространённых редакторов, используемых в мире UNIX, написанный на языке высокого уровня Elisp.

Выполнение лабораторной работы

1. Устанавливаю emacs.

```
vvazarcova@vvazarcova:~$ sudo dnf install emacs
Copr repo for gitflow owned by elegos          2.4 kB/s | 1.5 kB      00:00
Copr repo for browserpass owned by maksimbez  3.1 kB/s | 1.5 kB      00:00
Copr repo for l0sevk4 owned by peterwu        4.1 kB/s | 1.5 kB      00:00
Copr repo for l0sevk [                    ] --- B/s | 0 B      --:-- ETA
```

Рис. 1: Установка emacs

Открываю етас.

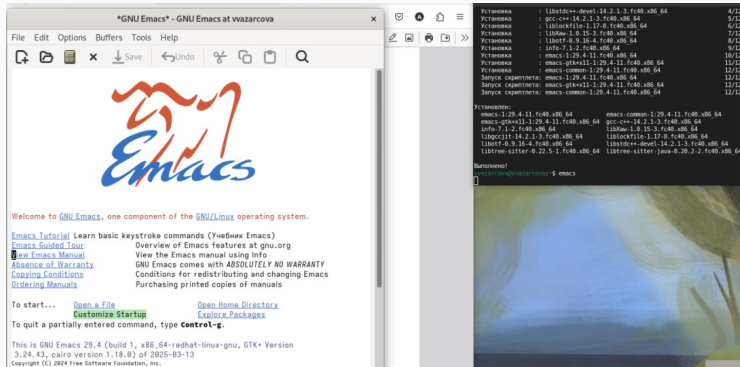


Рис. 2: Emacs

2. Создаю файл lab11.sh.

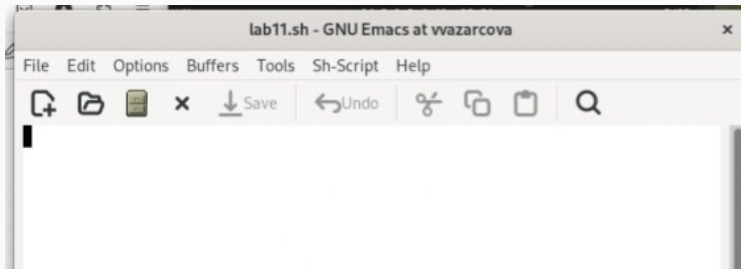
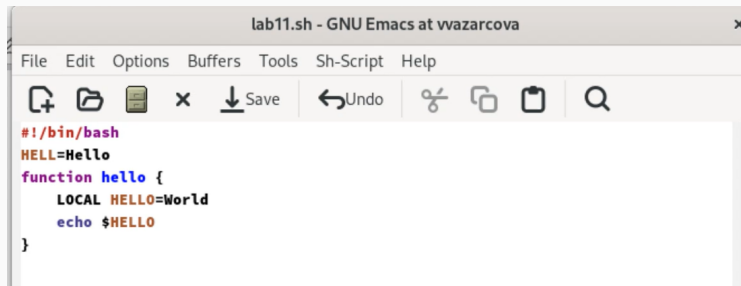


Рис. 3: Emacs

3. Набираю нужный текст.

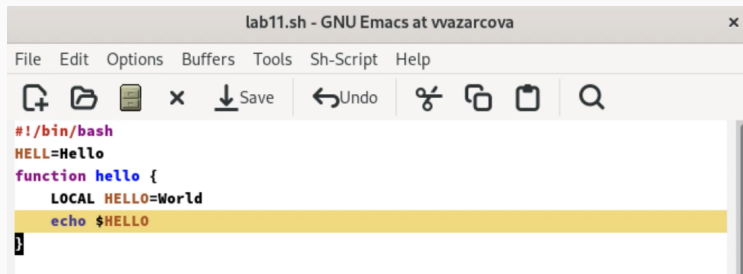


```
#!/bin/bash
HELL=Hello
function hello {
  LOCAL HELLO=World
  echo $HELLO
}
```

Рис. 4: Emacs

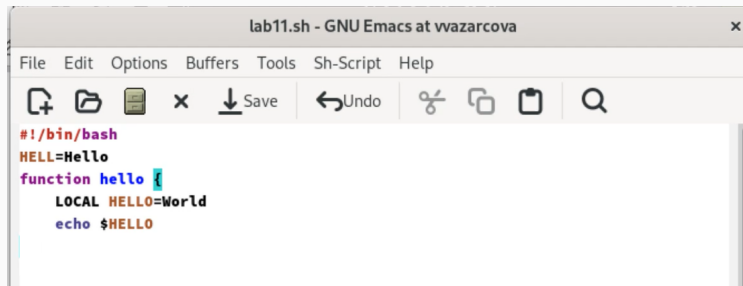
4. Сохраняю файл.

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-/).



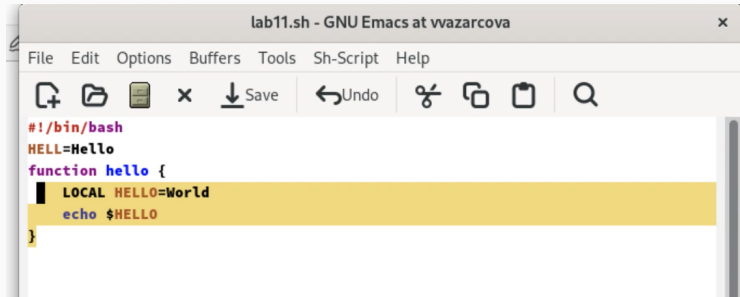
```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

Рис. 5: Шаг 1



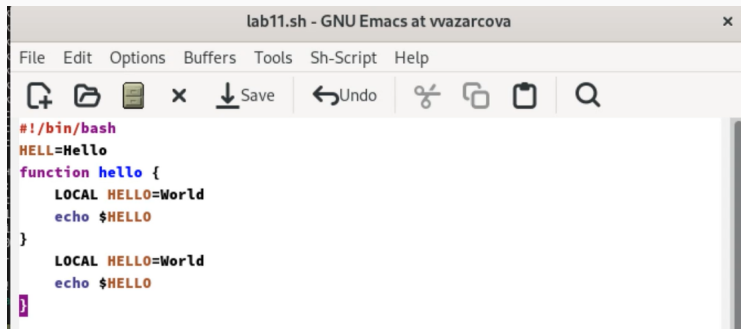
```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
```

Рис. 6: Шаг 2



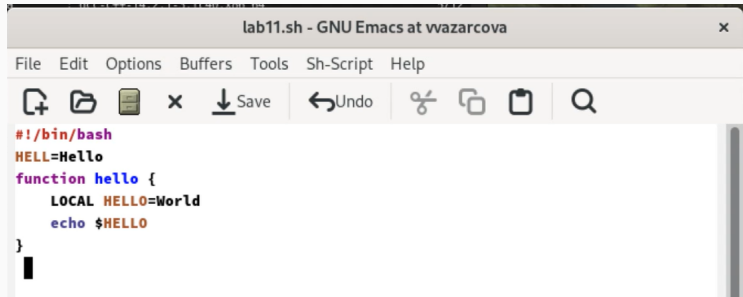
```
#!/bin/bash
HELLO=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

Рис. 7: Шаг 3



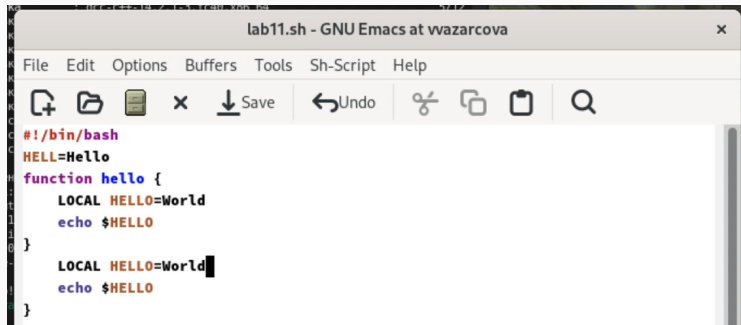
```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
LOCAL HELLO=World
echo $HELLO
}
```

Рис. 8: Шаг 4



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

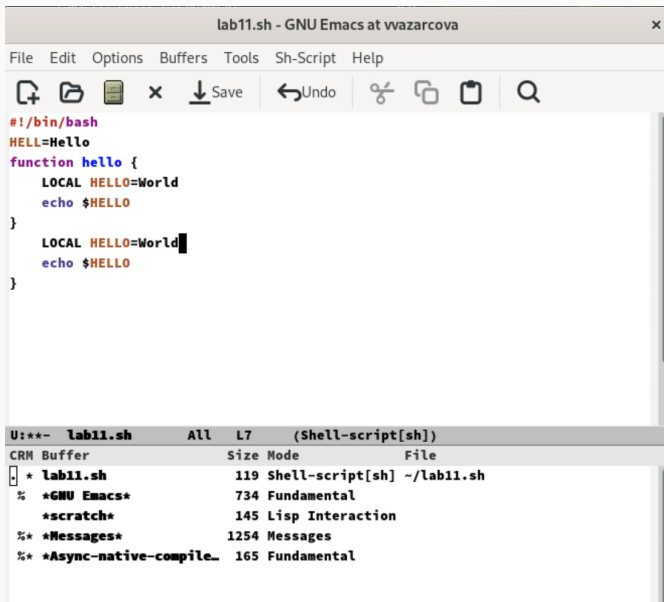
Рис. 9: Шаг 5



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

Рис. 10: Шаг 6

Выполнение лабораторной работы



The screenshot shows the GNU Emacs editor interface. The title bar reads "lab11.sh - GNU Emacs at wazarcova". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". The toolbar contains icons for file operations and editing. 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, the "Buffers" window is open, showing a list of buffers. The title bar of this window reads "U:*** lab11.sh All L7 (Shell-script[sh])". The list of buffers is as follows:

CRM	Buffer	Size	Mode	File
*	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:

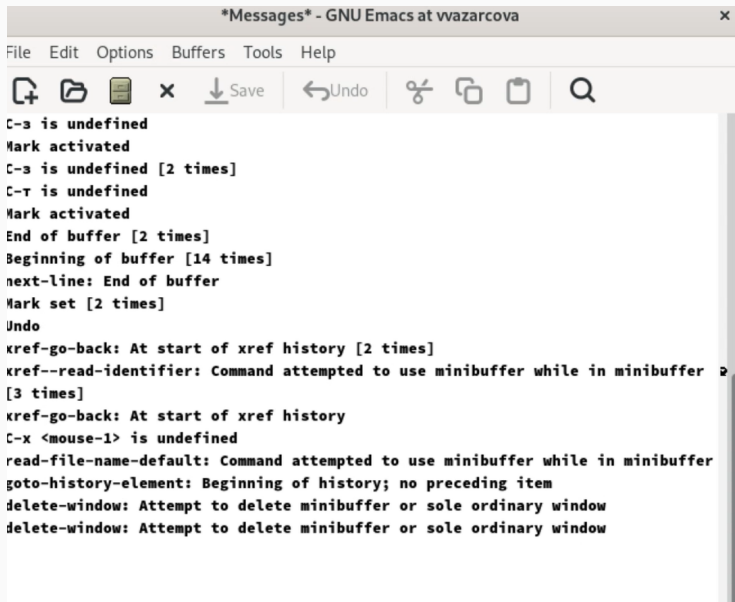
7.1 Вывести список активных буферов на экран (C-x C-b).

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



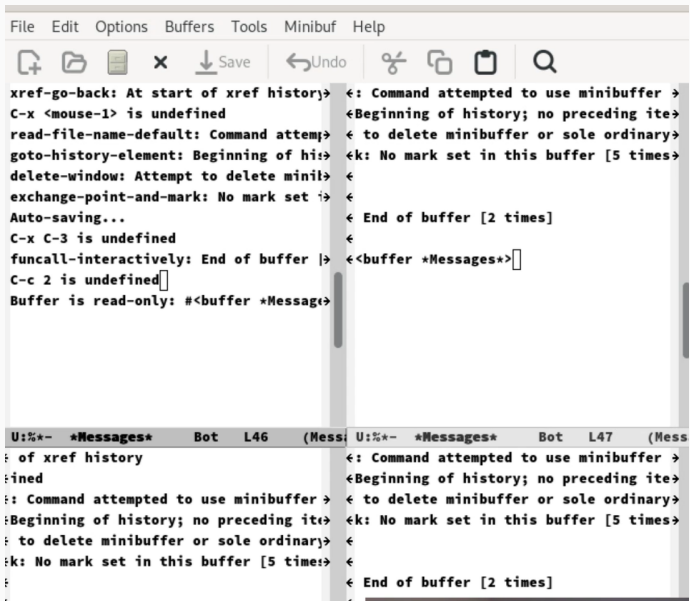
The screenshot shows the GNU Emacs Messages window titled "*Messages* - GNU Emacs at wazarcova". The window has a menu bar with "File", "Edit", "Options", "Buffers", "Tools", and "Help". Below the menu bar is a toolbar with icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search. The main area displays a list of error messages in a monospaced font.

```
*Messages* - GNU Emacs at wazarcova
File Edit Options Buffers Tools Help
[Icons: New, Open, Save, Close, Undo, Redo, Cut, Copy, Paste, Search]
C-3 is undefined
Mark activated
C-3 is undefined [2 times]
C-1 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

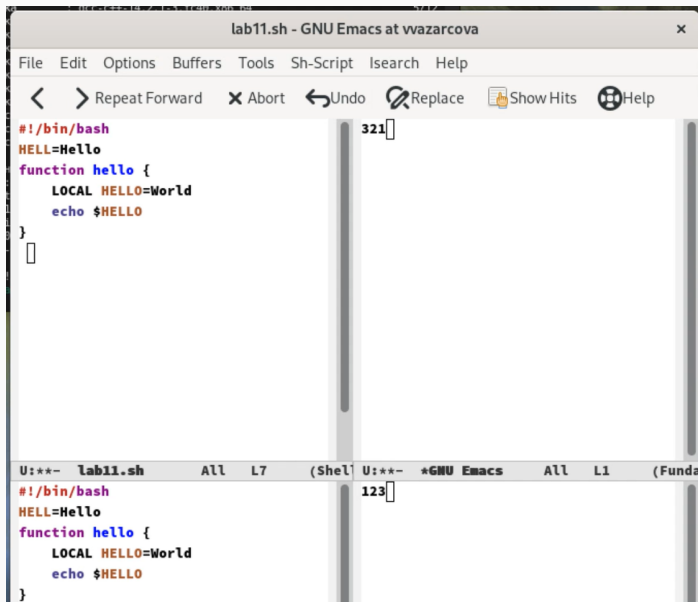


```
File Edit Options Buffers Tools Minibuf Help
[Icons: New, Open, Save, Close, Undo, Cut, Copy, Paste, Find]

xref-go-back: At start of xref history>
C-x <mouse-1> is undefined
read-file-name-default: Command attempted to use minibuffer>
goto-history-element: Beginning of history>
delete-window: Attempt to delete minibuffer>
exchange-point-and-mark: No mark set in this buffer [5 times]>
Auto-saving...
C-x C-3 is undefined
funcall-interactively: End of buffer |>
C-c 2 is undefined
Buffer is read-only: #<buffer *Messages*>

U:%*- *Messages* Bot L46 (Mess U:%*- *Messages* Bot L47 (Mess
: of xref history
: ined
: : Command attempted to use minibuffer>
: Beginning of history; no preceding item>
: to delete minibuffer or sole ordinary>
: k: No mark set in this buffer [5 times]>
:
: End of buffer [2 times]
```

Выполнение лабораторной работы



The image shows a GNU Emacs editor window titled "lab11.sh - GNU Emacs at vrazarcova". The editor contains a shell script with the following content:

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

```

The cursor is at the end of the script. Below the editor, a terminal window is open, showing the execution of the script. The terminal prompt is "U:***", and the output is "123".

```
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

Подводя итоги проведенной работе, я познакомилась с операционной системой Linux поглубже и получила практические навыки работы с редактором Emacs.

Если вам понравилось - посмотрите остальные мои презентации!