

Презентация по лабораторной работе №11

Текстовый редактор emacs

Джаллох Ишмаил

25 Апреля 2025

Российский университет дружбы народов, Москва, Россия

Информация

- Джаллох Ишмаил
- студентк НКА 05-24
- факультет физико-математических и естественных наук
- Российский университет дружбы народов
- 1032239614@rudn.ru
- <https://lsho-jah.github.io/ru/>

.. ..
.. ..

Цель работы

Получить практические навыки работы с редактором Emacs.

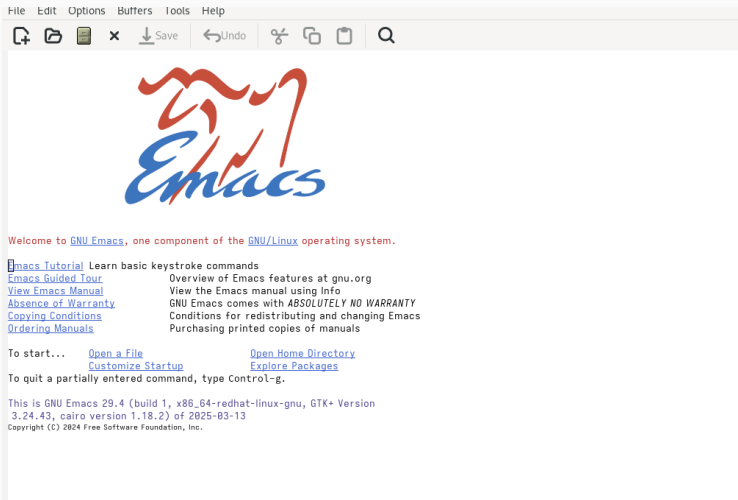
Для данной работы, мне надо была установить Emacs:

```
ijalloh@ijalloh:~$ sudo dnf install emacs
[sudo] password for ijalloh:
Updating and loading repositories:
Copr repo for browserpass owned by ma 100% | 507.0 B/s | 1.5 KiB |
Fedora 41 - x86_64 - Updates 100% | 8.7 KiB/s | 23.6 KiB |
```

Рис. 1: Установка Emacs

Основные команды с редактором

Выполнив Emacs в командной строке, я открыл текстовый редактор:



С помощью комбинации Ctrl-x Ctrl-f, создал файл lab07.sh:

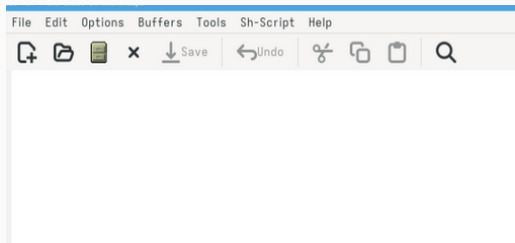
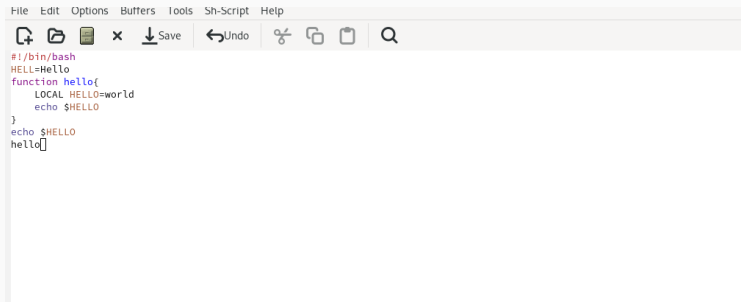


Рис. 3: Созданный файл

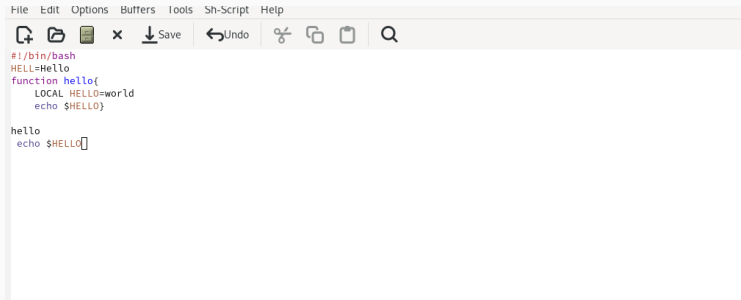
Я сохранил файл с помощью комбинации Ctrl-x Ctrl-s:

A screenshot of a text editor window. The title bar shows 'File Edit Options Buffers Tools Sh-Script Help'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The toolbar contains icons for opening, saving, undo, redo, 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
}
echo $HELLO
hello
```

Рис. 4: текст в lab07.sh

Вырезать целую строку (C-k):

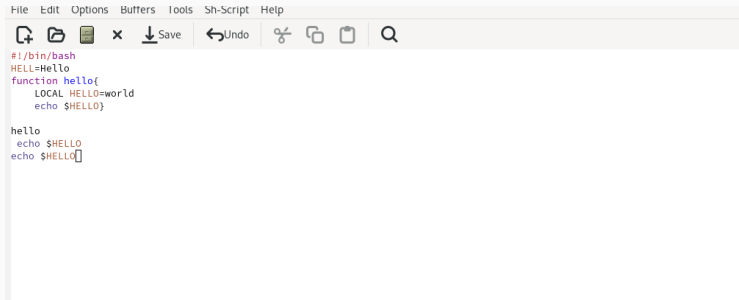


```
File Edit Options Buffers Tools Sh-Script Help
[Icons: New, Open, Save, Close, Undo, Cut, Copy, Paste, Find]
#!/bin/bash
HELL=Hello
function hello{
    LOCAL HELLO=world
    echo $HELLO}

hello
echo $HELLO
```

Рис. 5: Вырезание строки

С помощью C-у вставил эту строку в конец файла:

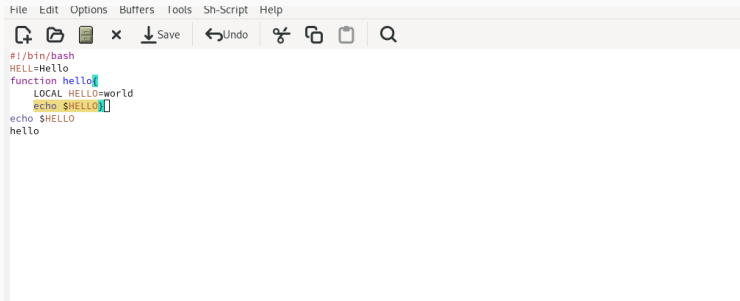


```
File Edit Options Buffers Tools Sh-Script Help
[Icons] Save Undo [Icons]
#!/bin/bash
HELL=Hello
function hello{
    LOCAL HELLO=world
    echo $HELLO}

hello
echo $HELLO
echo $HELLO
```

Рис. 6: Перемещение строку в конец файла

Выделил область текста (C-space):



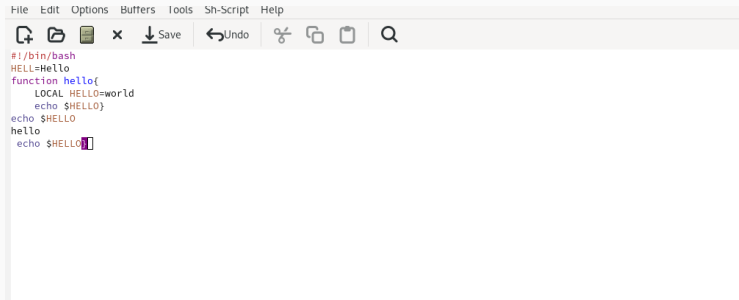
The image shows a screenshot of a text editor window. The window has a menu bar with the following items: File, Edit, Options, Buffers, Tools, Sh-Script, and Help. Below the menu bar is a toolbar with icons for opening a file, saving a file, closing a file, undo, redo, and search. The main text area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

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

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

Рис. 7: Выделенный текст

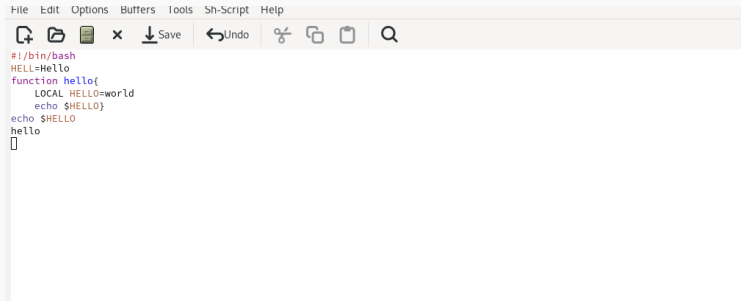
Скопировал область в буфер обмена (M-w) и вставил ее в конец файла:



```
File Edit Options Buffers Tools Sh-Script Help
[Icons: New, Open, Save, Close, Undo, Redo, Copy, Paste, Find]
#!/bin/bash
HELL=Hello
function hello{
    LOCAL HELLO=world
    echo $HELLO}
echo $HELLO
hello
echo $HELLO
```

Рис. 8: копирование и вставка

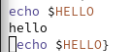
Выделил эту же область и на этот раз вырезал её (C-w):



```
File Edit Options Buffers Tools Sh-Script Help
[Icons: New, Open, Save, Close, Undo, Copy, Paste, Find]
#!/bin/bash
HELL=Hello
function hello{
    LOCAL HELLO=world
    echo $HELLO}
echo $HELLO
hello
█
```

Рис. 9: Вырезанная область

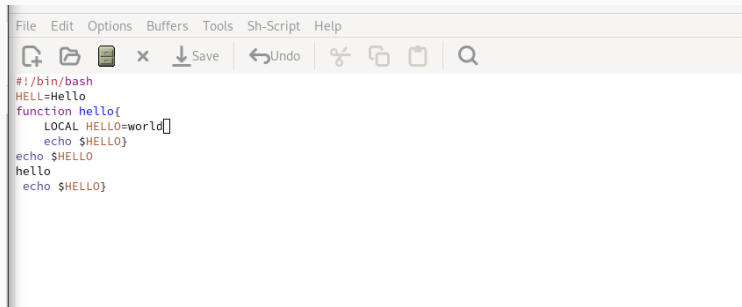
С помощью C-/ отменила последнее действие:



```
echo $HELLO  
hello  
[echo $HELLO}
```

Рис. 10: отмена действие

С помощью C-a можно переместить курсор в начало строки. С помощью C-e переместил курсор в конец строки:



```
File Edit Options Buffers Tools Sh-Script Help
[Icons: Open, Save, Undo, Cut, Copy, Paste, Find]
#!/bin/bash
HELL=Hello
function hello{
    LOCAL HELLO=world
    echo $HELLO}
echo $HELLO
hello
    echo $HELLO}
```

Рис. 11: Перемещение курсор в конец строки

Переместил курсор в начало и конец буфера с помощью M-< и M-> соответственно:

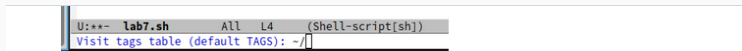
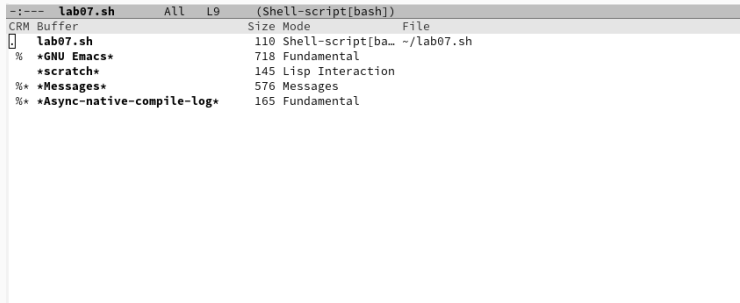


Рис. 12: Перемещение курсор в буфере

Выводил список активных буферов на экран с помощью C-x C-b:

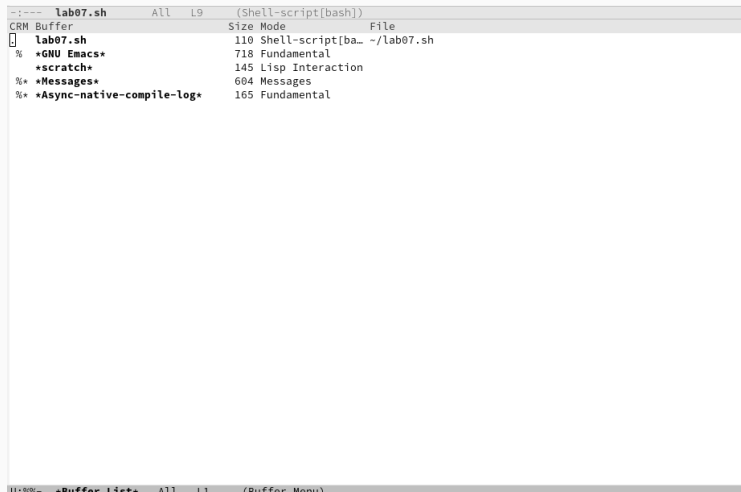
A screenshot of the Emacs buffer list window, which is titled "lab07.sh" and shows "All" buffers. The window displays a table of active buffers with columns for "CRM", "Buffer", "Size", "Mode", and "File". The buffers listed are "lab07.sh", "*GNU Emacs*", "*scratch*", "*Messages*", and "*Async-native-compile-log*".

| lab07.sh All L9 (Shell-script[bash]) | | | | |
|--------------------------------------|----------------------------|------|--------------------|------------|
| CRM | Buffer | Size | Mode | File |
| | lab07.sh | 110 | Shell-script[ba... | ~/lab07.sh |
| % | *GNU Emacs* | 718 | Fundamental | |
| | *scratch* | 145 | Lisp Interaction | |
| %* | *Messages* | 576 | Messages | |
| %* | *Async-native-compile-log* | 165 | Fundamental | |

Рис. 13: Активные буферы

Основные команды с редактором

С помощью C-x o переместился во вновь открытое окно со списком открытых буферов и переключился на другой буфер:



The screenshot shows the Emacs buffer list window, which is titled "lab07.sh" and has a subtitle "(Shell-script[bash])". The window displays a list of buffers with columns for "Buffer", "Size", "Mode", and "File". The buffers listed are:

| Buffer | Size | Mode | File |
|----------------------------|------|--------------------|------------|
| lab07.sh | 110 | Shell-script[ba... | ~/lab07.sh |
| *GNU Emacs* | 718 | Fundamental | |
| *scratch* | 145 | Lisp Interaction | |
| *Messages* | 604 | Messages | |
| *Async-native-compile-log* | 165 | Fundamental | |

С помощью C-x 0 закрыл окно со списком открытых буферов:



Рис. 15: Заккрытие окно

Без вывода списка буферов, я переключился между буферами:

A screenshot of a terminal window. The terminal has a light gray background. At the bottom, there is a dark gray status bar with white text that reads "U:%%- *Buffer List* All L6 (Buffer Menu)". Below the status bar, the prompt "Switch to buffer (default lab07.sh):" is displayed in blue text, followed by a cursor icon (a small vertical bar) indicating the current position for input.

```
U:%%- *Buffer List* All L6 (Buffer Menu)
Switch to buffer (default lab07.sh):
```

Рис. 16: Переключение между буферами

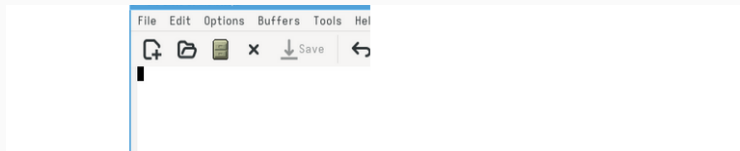
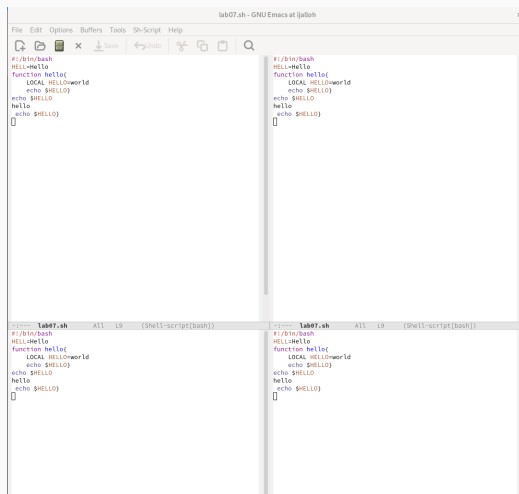


Рис. 17: Новый буфер

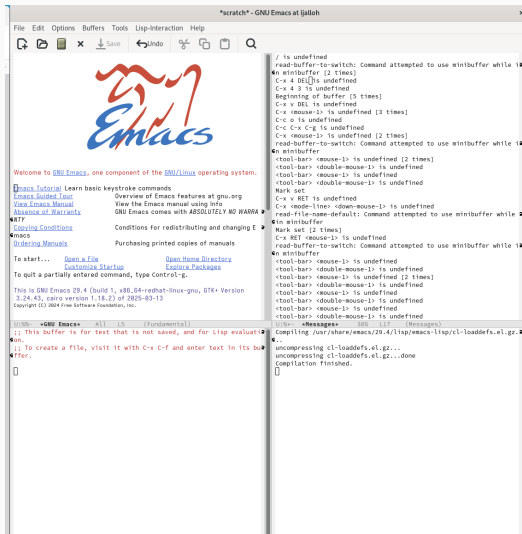
Основные кманды с редактором

Поделил фрейм на 4 части. Сначала я разделил фрейм на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по горизонтали (C-x 2):



Основные команды с редактором

В каждом из четырёх созданных окон открыл новый буфер:



Переключился в режим поиска (C-s) и искала Indent:

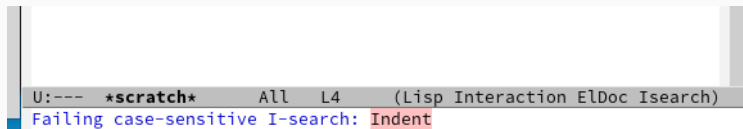
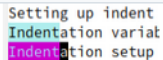


Рис. 20: Режим поиска

Переключался между результатами поиска, нажимая C-s и вышел из режима поиска, нажав C-g:



```
Setting up indent  
Indentation variat  
Indentation setup
```

Рис. 21: Переключение между результатами

Перешел в режим поиска и замены (M-% или M-x query replace):

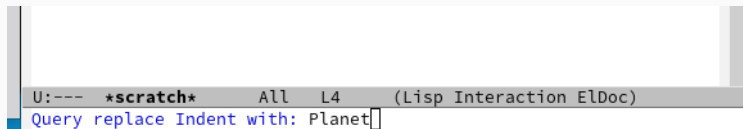


Рис. 22: Режим поиска

Нажав M-s o, я использовал другой режим поиска. Он отличается от предыдущего тем, что выводит результаты поиска в новом окне:

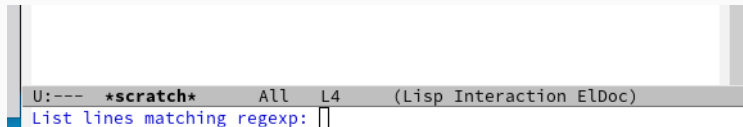
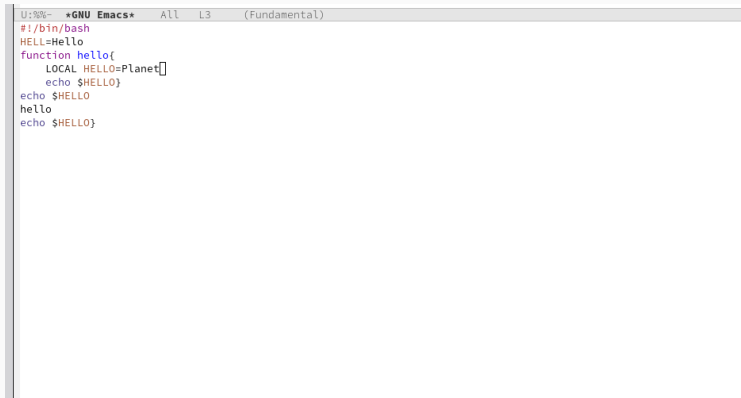


Рис. 23: другой режим поиска

Основные кманды с редактором



```
U:%%- *GNU Emacs* All L3 (Fundamental)
#!/bin/bash
HELL=Hello
function hello{
    LOCAL HELLO=Planet[]
    echo $HELLO}
echo $HELLO
hello
echo $HELLO}
```

Рис. 24: Результаты поиска

Выводы

При выполнении данной работы я получил практические навыки работы с Emacs.