

Научная презентация по лабораторной работе №11

Дисциплина: Операционные системы

Абрамова У. М.

25.04.2025

Российский университет дружбы народов, Москва, Россия

Математический институт имени Никольского, Москва, Россия

Информация

- Абрамова Ульяна Михайловна
- Студент НММбд-01-24
- Российский университет дружбы народов
- 1132246782@pfur.ru
- <https://github.com/Ulyana-abr>

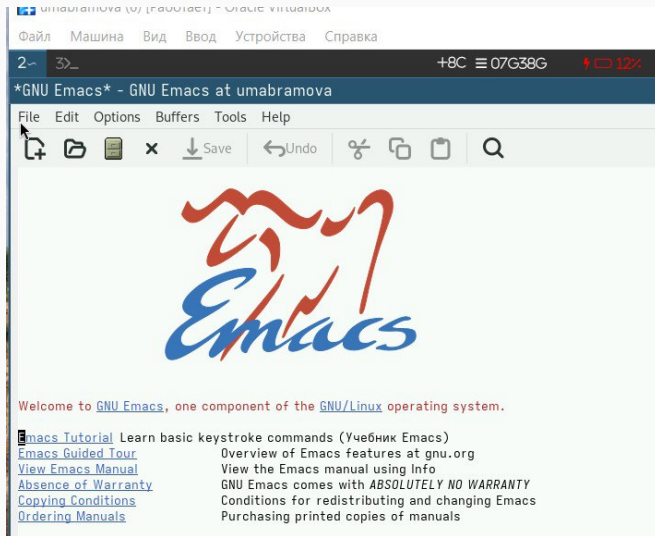
Вводная часть

- Работа с Emacs остаётся актуальной, несмотря на появление новых инструментов и редакторов каждый год, благодаря своей настраиваемости, интеграции с другими инструментами и работы на разных платформах

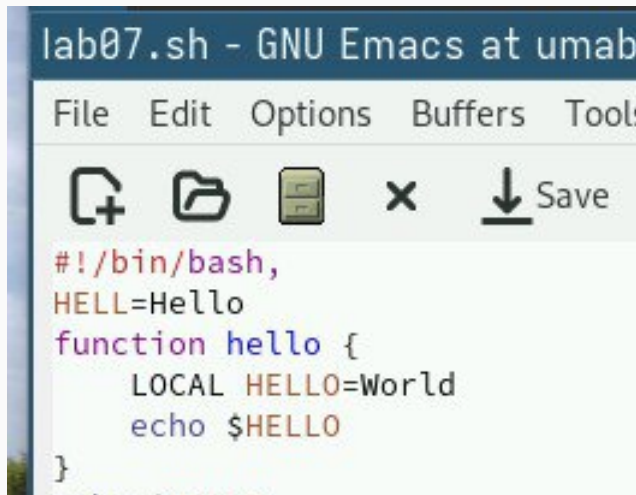
- Познакомиться с операционной системой Linux.
- Получить практические навыки работы с редактором Emacs

- linux
- emacs

Открываю emacs (рис. 1).



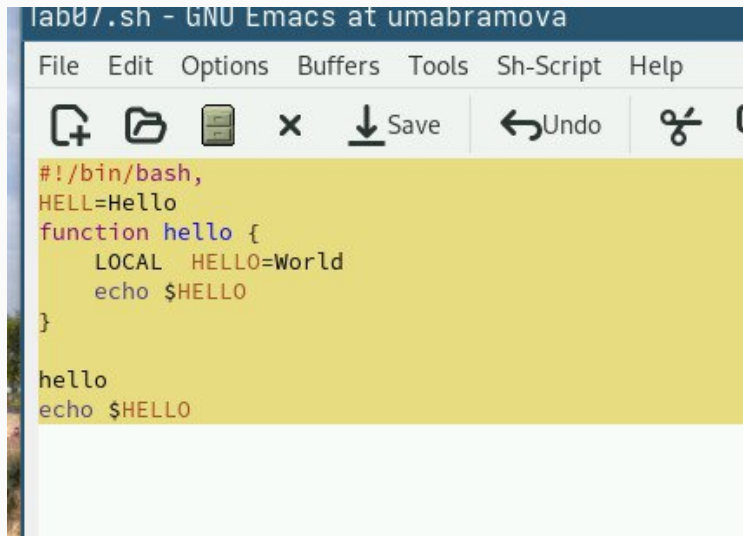
Создаю файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f и набираю необходимый текст (рис. 2)

A screenshot of the GNU Emacs editor window. The title bar at the top reads "lab07.sh - GNU Emacs at umab". Below the title bar is a menu bar with "File", "Edit", "Options", "Buffers", and "Tools". Under the "File" menu, there are icons for creating a new file, opening a file, saving a file, and closing a file, followed by a "Save" button with a downward arrow icon. The main text area contains a shell script with syntax highlighting: a red shebang line "#!/bin/bash,", an orange variable assignment "HELL=Hello", a purple function definition "function hello {", an indented orange variable assignment "LOCAL HELLO=World", an indented orange command "echo \$HELLO", and a closing purple brace "}".

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

Сохраняю файл с помощью комбинации Ctrl-x Ctrl-s.

Далее проделываю с текстом стандартные процедуры редактирования, осуществленное комбинацией клавиш. Вырезаю командой “C-k” целую строку и вставляю ее в конец файла (C-y), после чего выделяю текст благодаря “C-space” (рис. 3)



The image shows a screenshot of the GNU Emacs editor window. The title bar at the top reads "tab07.sh - GNU Emacs at umabramova". Below the title bar is a menu bar with the following items: "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". Underneath the menu bar is a toolbar containing several icons: a window icon, a folder icon, a document icon, a close icon (X), a save icon (downward arrow), an undo icon (curved arrow), and a redo icon (curved arrow). The main editing area has a yellow background and contains the following text:

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

Рис. 3: изменение текста

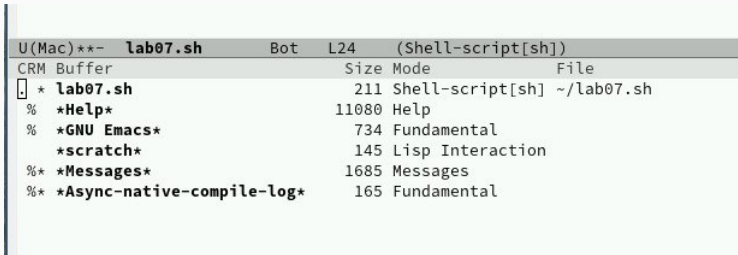
Скопировав область в буфер обмена с помощью “М-w”, вставляю область в конец файла.
(рис. 4)

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

Вновь выделяю эту область и на этот раз вырезаю её, используя “C-w”. Отменяю последнее действие благодаря “C-/”

Далее учусь использовать команды по перемещению курсора: - перемещение курсора в начало строки - “С-а” - перемещение курсора в конец строки - “С-е” - перемещение курсора в начало буфера - “М-<” - перемещение курсора в конец буфера - “М->”

Вывожу список активных буферов на экран (C-x C-b). (рис. 5)

A screenshot of the Emacs buffer list window, which is titled 'U(Mac)**~ lab07.sh Bot L24 (Shell-script[sh])'. The window shows a list of active buffers with columns for 'CRM Buffer', 'Size', 'Mode', and 'File'. The buffers listed are: '* lab07.sh' (211 bytes, Shell-script[sh], ~/lab07.sh), '*Help*' (11080 bytes, Help), '*GNU Emacs*' (734 bytes, Fundamental), '*scratch*' (145 bytes, Lisp Interaction), '*Messages*' (1685 bytes, Messages), and '*Async-native-compile-log*' (165 bytes, Fundamental).

U(Mac)**~ lab07.sh Bot L24 (Shell-script[sh])			
CRM Buffer	Size	Mode	File
* lab07.sh	211	Shell-script[sh]	~/lab07.sh
% *Help*	11080	Help	
% *GNU Emacs*	734	Fundamental	
scratch	145	Lisp Interaction	
%% *Messages*	1685	Messages	
%% *Async-native-compile-log*	165	Fundamental	

Рис. 5: вывод активных буферов

Перемещаюсь во вновь открытое окно (C-x o) со списком открытых буферов и переключитесь на другой буфер. (рис. 6)

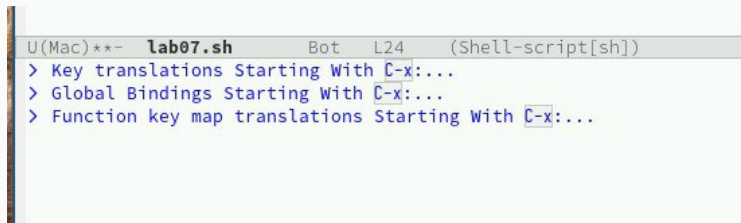


Рис. 6: переключение на другой буфер

Закрываю это окно (C-x 0). (рис. 7)

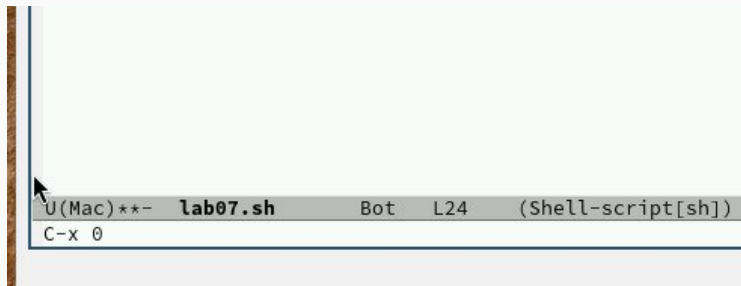


Рис. 7: закрытие окна

Теперь вновь переключаюсь между буферами, но уже без вывода их списка на экран (С-х b)

Разделяю фрейм на два окна по вертикали (С-х 3) (рис. 8) , а затем каждое из этих окон на две части по горизонтали (С-х 2) (рис. 9)

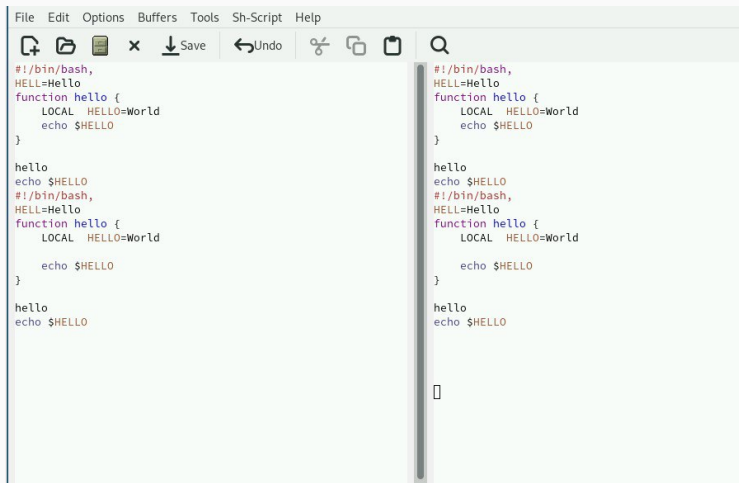


Рис. 8: разделение фрейма

```
ho $HELLO

HELLO

**~ lab07.sh Bot L24 (Shell-script[sh])
/bash,
ello
on hello {
CAL HELLO=World

ho $HELLO

HELLO

U(Mac)*lab07.sh Bot L22 (Shell-scri
#!/bin/bash,
HELL=Hello
function hello {
LOCAL HELLO=World

echo $HELLO
}

hello
echo $HELLO

[]
```

Рис. 9: разделение фрейма

В каждом из четырёх созданных окон открываю новый буфер (рис. 10)

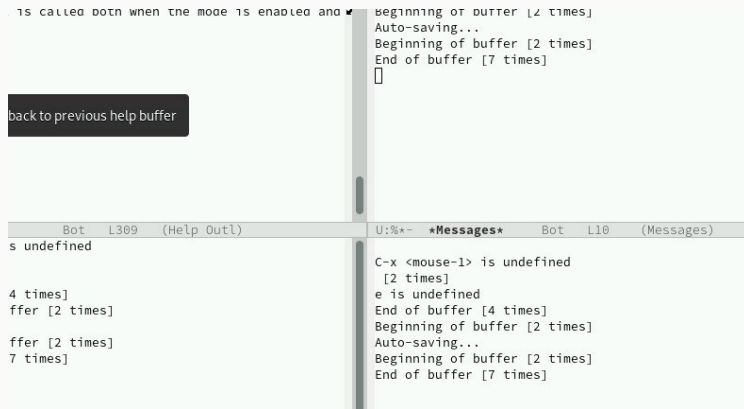


Рис. 10: открытие буфера

Переключаюсь в режим поиска (C-s) и нахожу несколько слов, присутствующих в тексте. (рис. 11)

A screenshot of a terminal window with a light green background. The terminal shows a shell script being edited. The script starts with a shebang line, followed by a variable assignment, a function definition, and a call to the function. The words 'Hello', 'HELLO', and 'hello' are highlighted in cyan, magenta, and magenta respectively. A mouse cursor is visible at the bottom left of the terminal window.

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


Переключаюсь между результатами поиска, нажимая C-s. Выхожу из режима поиска, нажав C-g. Перехожу в режим поиска и замены (M-%), ввожу текст, который следует найти и заменить, нажимаю Enter , затем ввожу текст для замены. После того как подсветились результаты поиска, нажимаю ! для подтверждения замены. (рис. 12)

```
U(Mac)**- lab07.sh Bot L19 (Shell-script[sh])
```

```
HELL=Hello
function hello {
    LOCAL HELLO=World

    echo $HELLO
}
```

```
hello
ec summ█
```

```
U(Mac)**- lab07.sh Bot L19 (Shell-script[sh])
```

```
Replaced 1 occurrence
```

Испробую другой режим поиска, нажав M-s o, который отличается от обычного нахождением файла, а не фрагмента текста.

В данной работе мы познакомились с еще одним редактором операционной системой Linux. Получили практические навыки работы с редактором Emacs.

- Чем больше вы практикуетесь и пробуете новое, тем больше и чаще вы будете получать знаний и опыта в разных сферах.