

# Лабораторная работа №9

---

Кочкарев “sakochkarev” Станислав

RUDN University

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

- Ознакомление с теоретическим материалом
- Ознакомление с редактором emacs
- Выполнение упражнений с редактором emacs

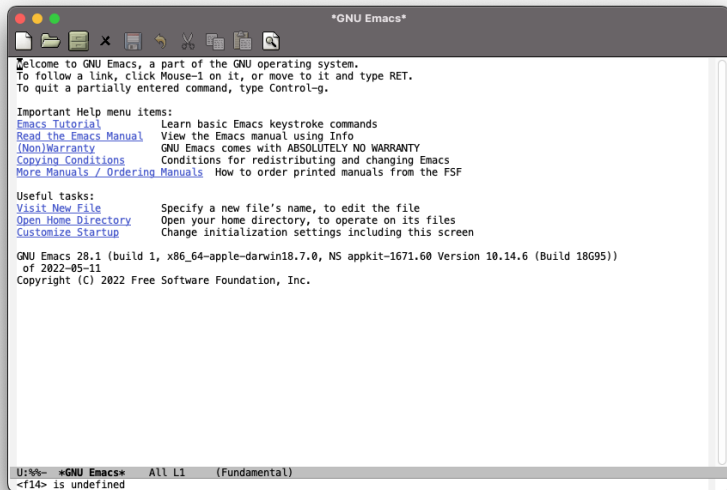
## Выполнение лабораторной работы

---

Предварительно был установлен текстовый редактор  
emacs.

Первым делом были изучены теоретические материалы и было произведено ознакомление с редактором etacs. После этого мы перешли к выполнению упражнений.

Сперва-наперво мы открыли сам редактор emacs.

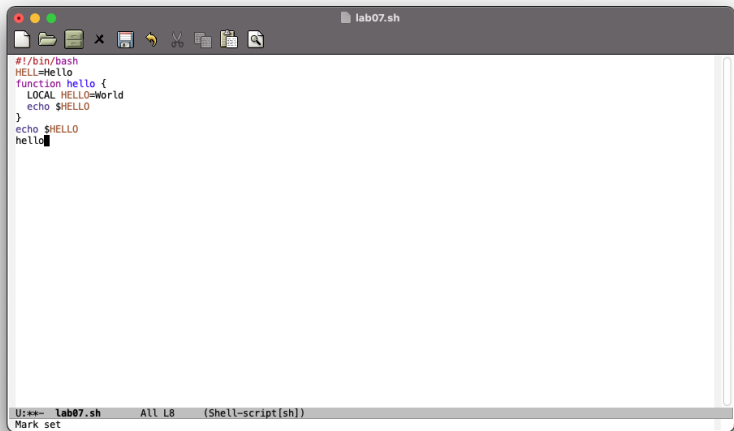


Далее был создан файл `lab07.sh` с помощью приведенной комбинации клавиш.





После этого в открытом файле был написан приведенный текст.



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

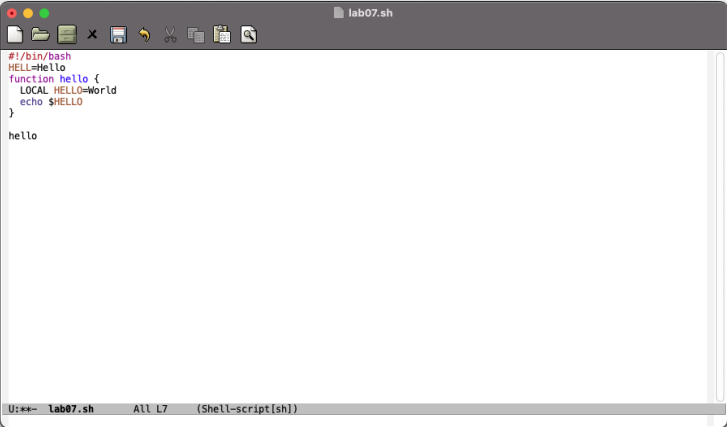
U:\*~ lab07.sh All L8 (Shell-script[sh])  
Mark set

По окончании записи файла мы сохранили его с помощью комбинации клавиш.

```
U:--- lab07.sh All L8 (Shell-script[sh])  
Wrote /Users/squidass/lab07.sh
```

Далее шли упражнения по стандартным процедурам редактирования.

Мы вырезали одной командой целую строку и вставили ее в конец файла.



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

The screenshot shows a macOS terminal window with a dark title bar containing the text 'lab07.sh'. The terminal window has a standard macOS toolbar with icons for file operations. The main area of the terminal displays a shell script with syntax highlighting: a shebang line, a variable assignment, a function definition with a local variable, and a function call. Below the script, the output 'hello' is visible. At the bottom, the terminal's status bar shows the prompt 'U:~-- lab07.sh', the file size 'All L7', and the shell type '(Shell-script[sh])'.

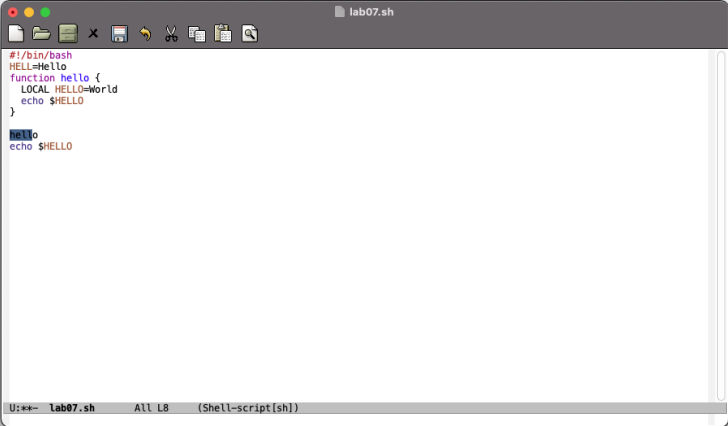


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

hello
echo $HELLLO
```

U:\*~ lab07.sh All L9 (Shell-script[sh])  
Mark set

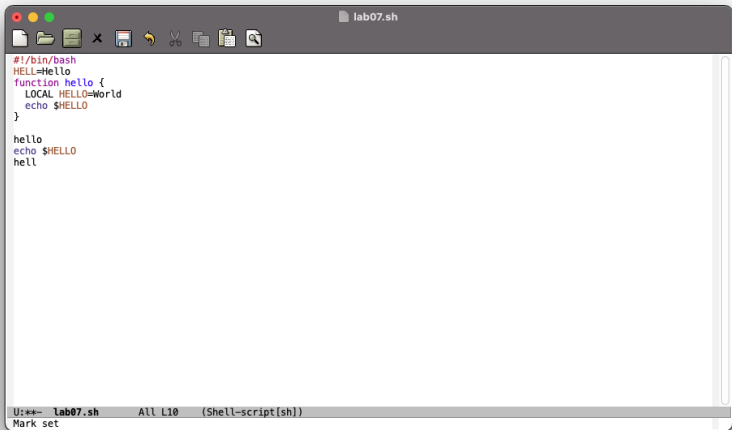
После мы выделили область текста, скопировали эту область в буфер обмена и вставили ее в конец файла.



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

hello
echo $HELLO
```

The screenshot shows a macOS terminal window with a dark title bar containing the text 'lab07.sh'. The terminal has a standard macOS menu bar at the top with icons for file operations (new, open, save, print, copy, paste, find) and window management (close, zoom, full screen). The terminal content displays a shell script with a function 'hello' that prints 'World' when called. Below the script, the command 'hello' has been executed, and the output 'World' is visible. The status bar at the bottom of the terminal shows the prompt 'U:~\$' followed by 'lab07.sh', 'All L8', and '(Shell-script[sh])'.



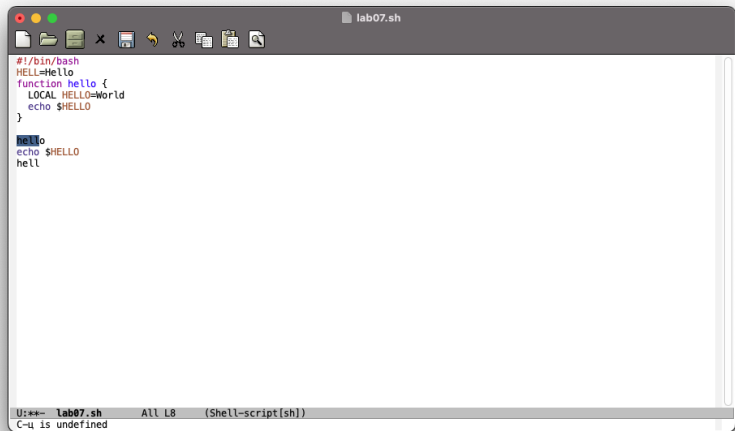
The image shows a terminal window with a dark title bar containing standard macOS window controls (red, yellow, green buttons) and a title 'lab07.sh'. Below the title bar is a toolbar with icons for file operations like opening, saving, and searching. The main area of the window contains a shell script with the following content:

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

hello
echo $HELLO
hell
```

The script defines a variable `HELL` with the value `Hello`, a function `hello` that sets a local variable `HELLO` to `World` and prints it, and then calls `hello`, prints the global `HELL` value, and finally prints `hell`. The bottom status bar of the window shows the prompt `U:~$`, the filename `lab07.sh`, the window title `All L10`, and the shell type `(Shell-script[sh])`. Below the status bar, the text `Mark set` is visible.

В конце мы вновь выделили эту область и вырезали ее.

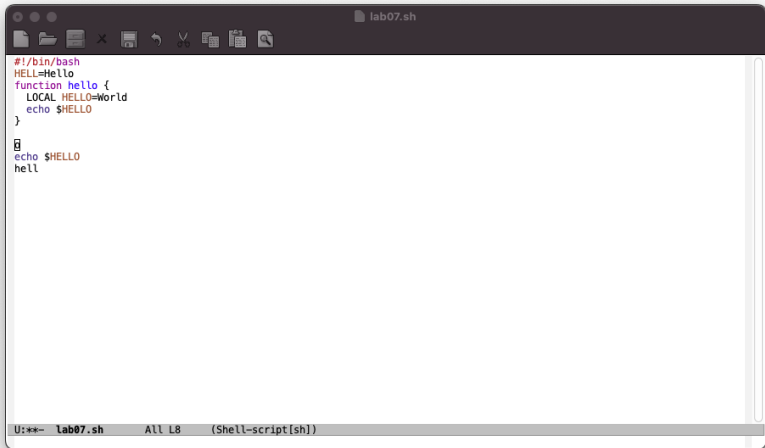


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

hello
echo $HELLO
hell
```

U:~-- lab07.sh All L8 (Shell-script[sh])  
C-u is undefined



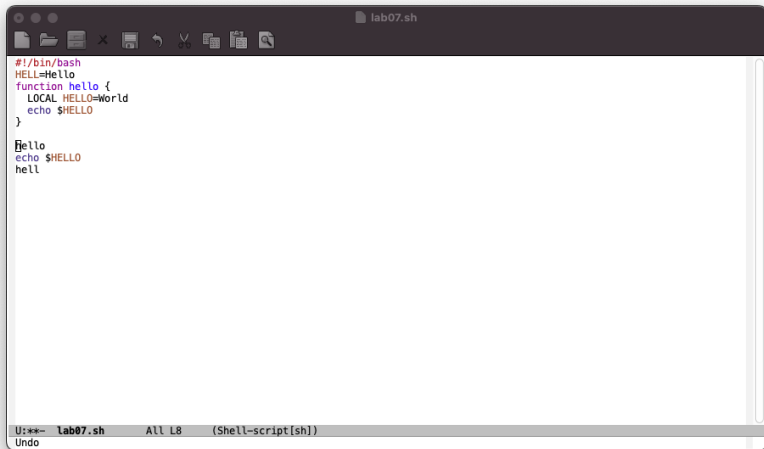


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

echo $HELLO
hell
```

U:~-- lab07.sh All L8 (Shell-script[sh])

Последнее действие мы отменили.



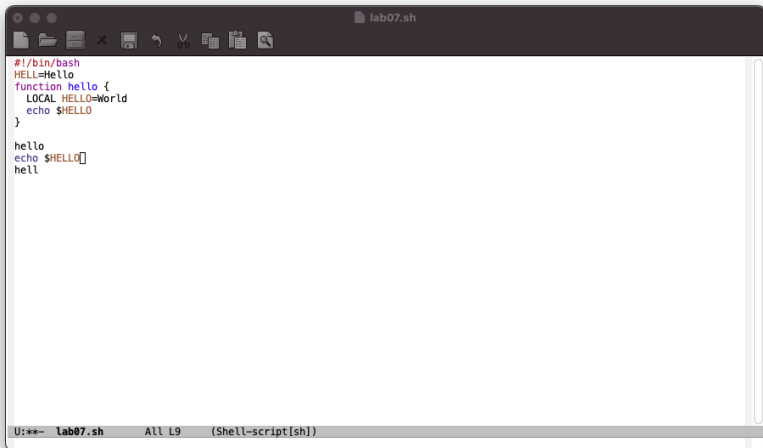
```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

Hello
echo $HELLO
hell
```

U:\*\*\* lab07.sh All L8 (Shell-script[sh])  
Undo

После шли упражнения по перемещению курсора.

Сначала мы переместили курсор в начало строки, после этого в конец строки.

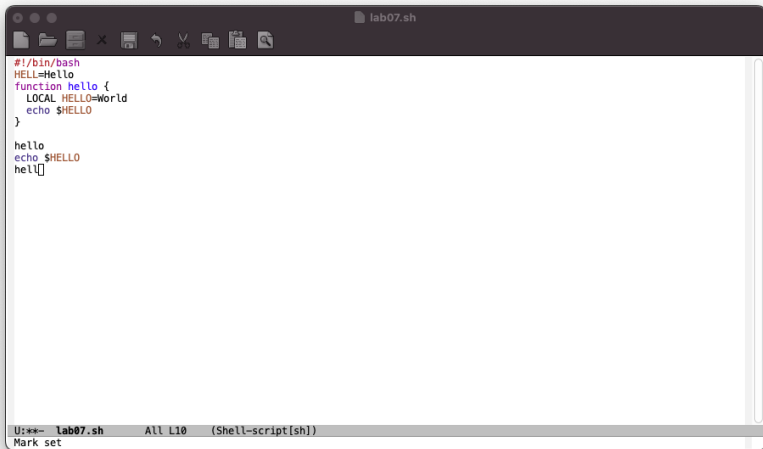


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

hello
echo $HELLO
hell
```

U:\*\*\*- lab07.sh All L9 (Shell-script[sh])

А также переместили курсор в начало буфера и в конец.



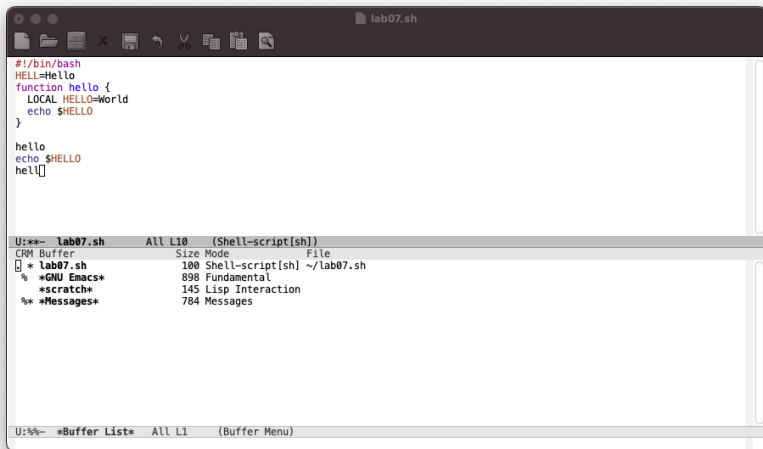
```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

hello
echo $HELLO
hell
```

U:\*\*\*- lab07.sh All L10 (Shell-script[sh])  
Mark set

Следующими шли задания по управлению буферами.

Первым делом мы вывели список активных буферов на экран.



The screenshot shows an Emacs window titled "lab07.sh". The main buffer contains the following text:

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

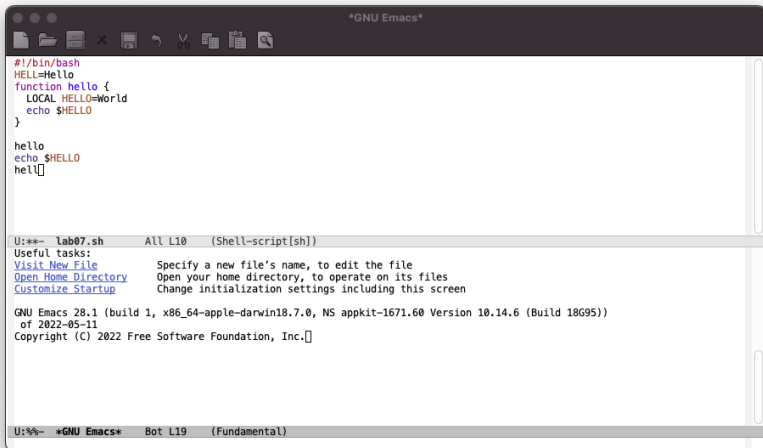
hello
echo $HELLO
hell
```

Below the main buffer, the "Buffer List" is displayed, showing a table of active buffers:

U:*	lab07.sh	All	L10	(Shell-script[sh])	
CRM	Buffer		Size	Mode	File
	* lab07.sh		100	Shell-script[sh]	~/lab07.sh
%	*GNU Emacs*		898	Fundamental	
	*scratch*		145	Lisp Interaction	
%*	*Messages*		784	Messages	

At the bottom of the window, the status bar shows: "U:%%- \*Buffer List\* All L1 (Buffer Menu)".

Далее сочетанием клавиш переместились на открытое окно со списком открытых буферов и в нем переключились на другой буфер.



The screenshot shows the GNU Emacs editor window titled '\*GNU Emacs\*'. The main editing area contains a shell script with the following content:

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

hello
echo $HELLO
hell
```

Below the script, the output of the commands is visible:

```
hello
World
hell
```

The status bar at the bottom of the window displays the following information:

U:\*\*\* lab07.sh All L10 (Shell-script[sh])

Useful tasks:

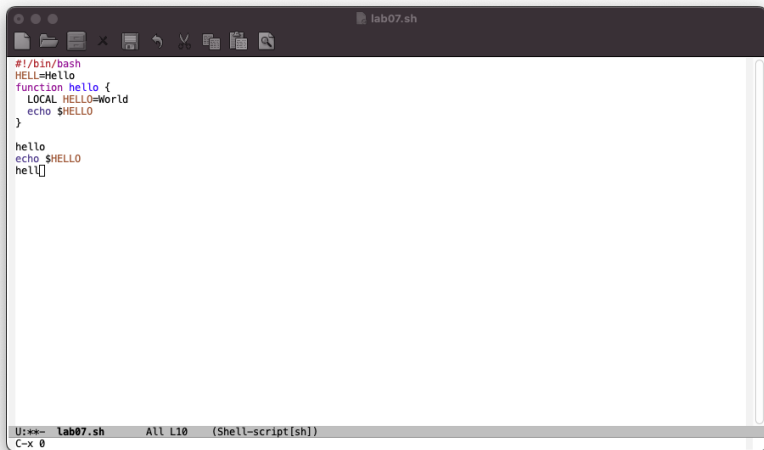
- [Visit New File](#) Specify a new file's name, to edit the file
- [Open Home Directory](#) Open your home directory, to operate on its files
- [Customize Startup](#) Change initialization settings including this screen

GNU Emacs 28.1 (build 1, x86\_64-apple-darwin18.7.0, NS appkit-1671.60 Version 10.14.6 (Build 18G95))  
of 2022-05-11  
Copyright (C) 2022 Free Software Foundation, Inc.

U:\*\*\* \*GNU Emacs\* Bot L19 (Fundamental)



После этого мы закрыли окно с только что открытым буфером.

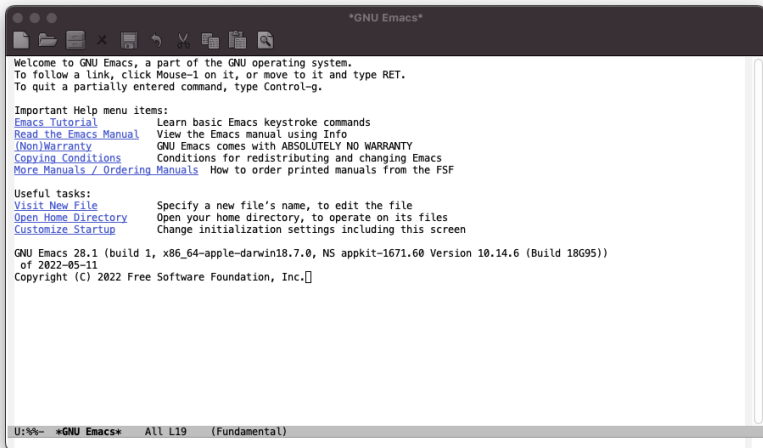


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

hello
echo $HELLO
hell
```

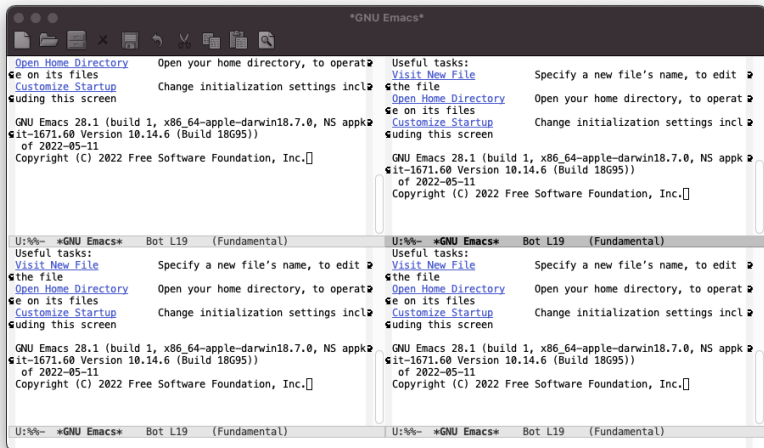
The screenshot shows a terminal window with a dark title bar containing the text 'lab07.sh'. The terminal has a standard macOS-style toolbar with icons for file operations. The script content is color-coded: blue for the shebang, red for variable assignments, purple for function definitions, and orange for echo commands. The script defines a function 'hello' that sets a local variable 'HELLO' to 'World' and prints it. Below the function definition, the word 'hello' is printed, followed by the command 'echo \$HELLO' which prints 'Hello', and then 'hell' followed by a cursor. The bottom status bar shows the prompt 'U:\*\*\*', the filename 'lab07.sh', the line number 'All L10', and the shell type '(Shell-script[sh])'.

В конце мы также переключились между буферами, однако теперь без использования окна, а только используя сочетание клавиш.

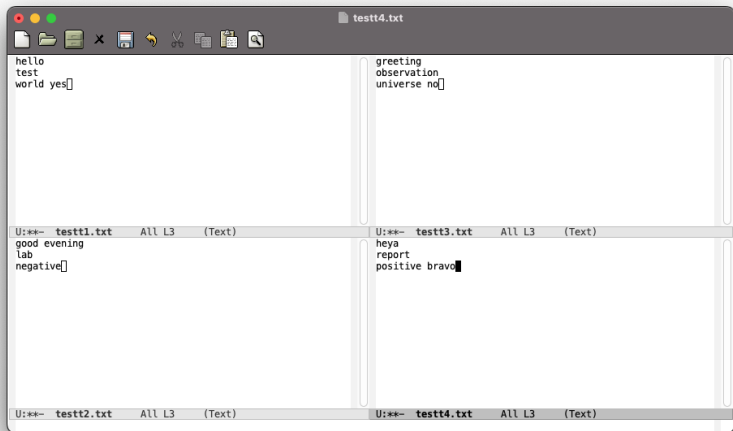


Предпоследним блоком заданий были задания по управлению окнами.

Первым делом сочетаниями клавиш мы поделили фрейм на 4-е части:  
два окна по вертикали и два по горизонтали.

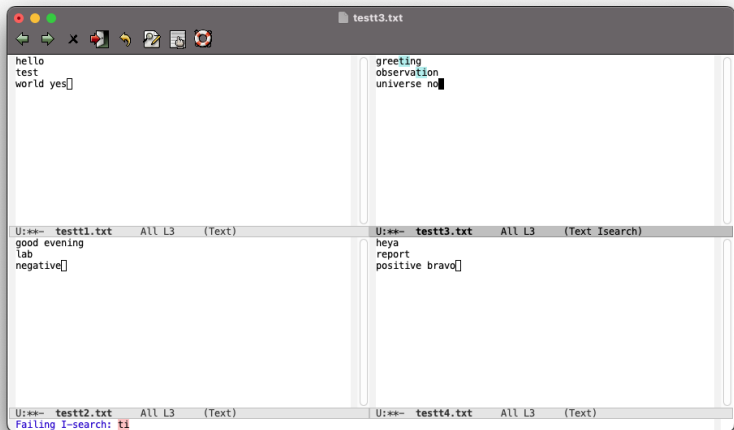


После этого в каждом из созданных окон мы создали новый файл и напечатали в нем несколько строк текста.

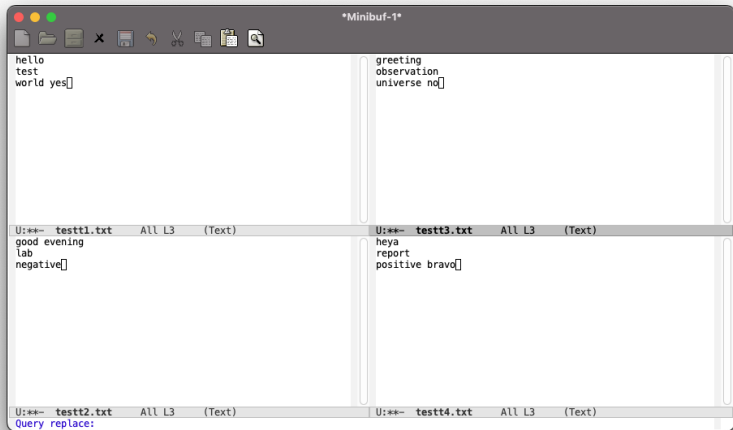


Последним блоком заданий были задания связанные с режимом поиска.

Для начала мы переключились в обычный режим поиска и нашли несколько слов в тексте. По результатам можно было переключаться сочетанием клавиш.

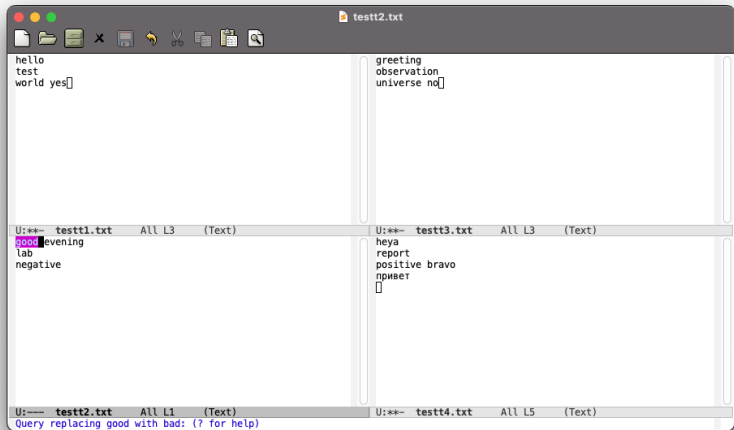


Далее мы вышли из этого режима поиска и перешли в режим замены текста.

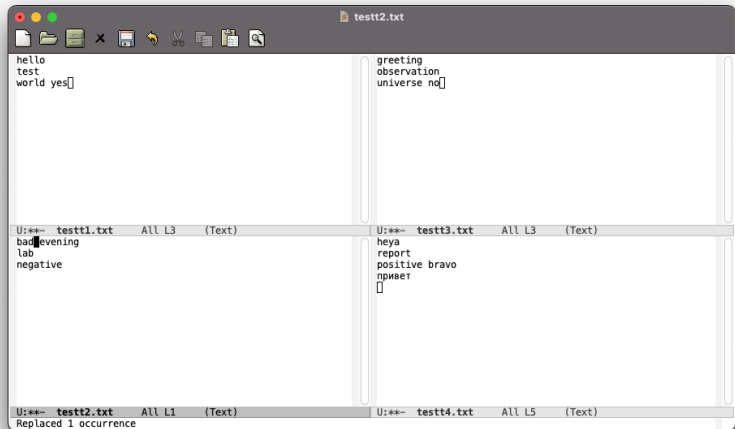




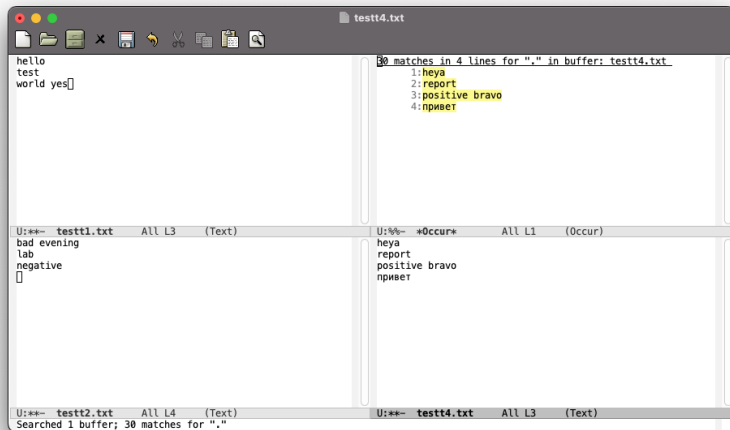
В этом режиме первым делом мы ввели текст, который следует найти и заменить, нажали Enter и затем ввели текст для замены.



В буфере подсветился текст для замены и для подтверждения замены мы нажали !.



В конце мы испробовали другой режим поиска, нажав M-s o.



Данный режим является режимом поиска с использованием регулярных выражений.

По выполнении лабораторной работы мы получили практические навыки работы с редактором emacs, а также немного дополнительно познакомились с операционной системой Linux.