

# Отчет по лабораторной работе №10

---

Безрук Мария Андреевна

RUDN University, Moscow, Russian Federation

## ##Цель работы

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

## Задание

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

# Выполнение лабораторной работы

- 1) Откроем редактор Emacs с помощью команды «emacs&».

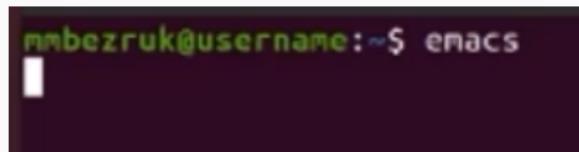


Figure 1: Открытие редактора

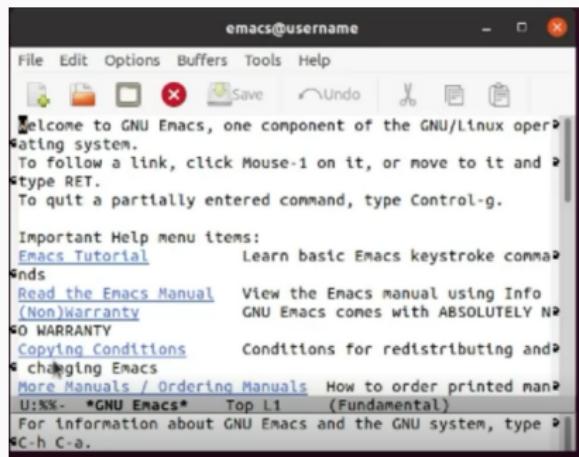


Figure 2: Редактор

# Выполнение лабораторной работы

2) Создадим файл lab10.sh с помощью комбинации «Ctrl-x»«Ctrl-f».

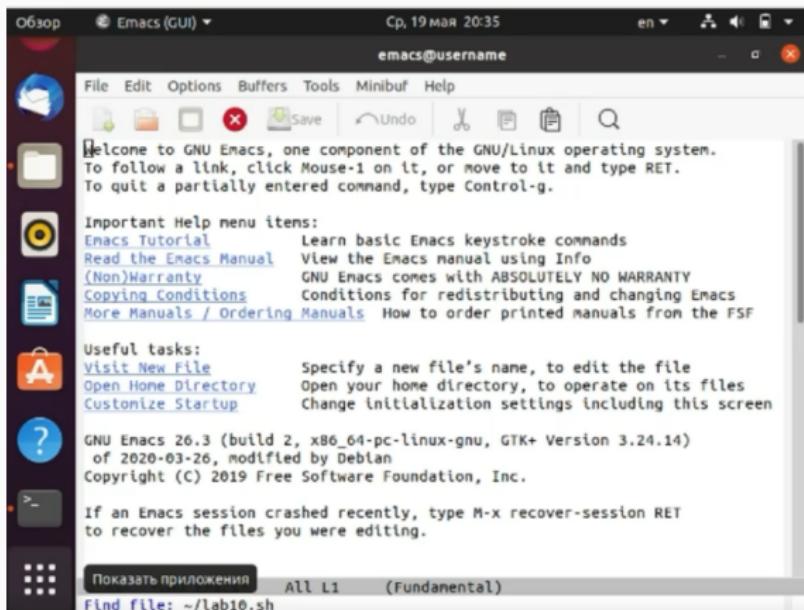
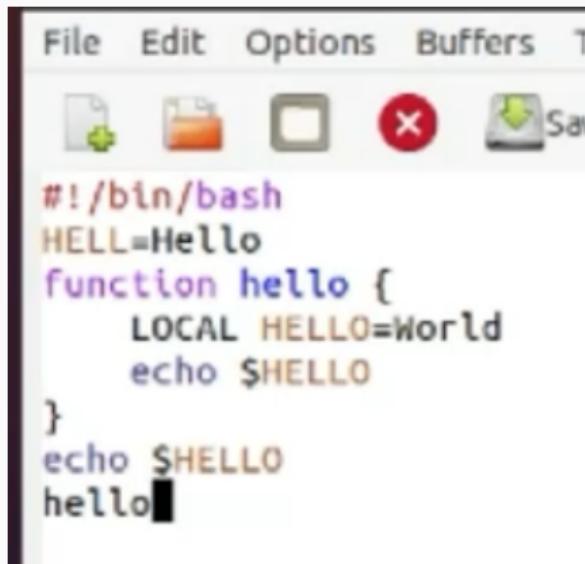


Figure 3: Создание файла

## Выполнение лабораторной работы

- 3) В открывшемся буфере наберем необходимый текст



The screenshot shows a terminal window with a menu bar containing 'File', 'Edit', 'Options', 'Buffers', and a number '1'. Below the menu is a toolbar with icons for file operations: a document with a green plus sign, a folder, a square, a red circle with a white 'X', and a save icon. The main area of the terminal contains the following shell script code:

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

Figure 4: Набор текста

## Выполнение лабораторной работы

- 4) Сохраним файл с помощью комбинации «Ctrl-x»«Ctrl-s».
- 5.1) Вырежем одной командой целую строку («Ctrl-k»)



The screenshot shows a terminal window with a menu bar containing 'File', 'Edit', 'Options', 'Buffers', and 'Tools'. Below the menu are several icons: a file with a green plus sign, a file with an orange minus sign, a square with a minus sign, a red circle with a white 'X', and a save icon with a green arrow. The main area of the terminal contains the following Bash script code:

```
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
```

Figure 5: Вырезка строки

## Выполнение лабораторной работы

5.2) Вставим эту строку в конец файла («Ctrl-y»)

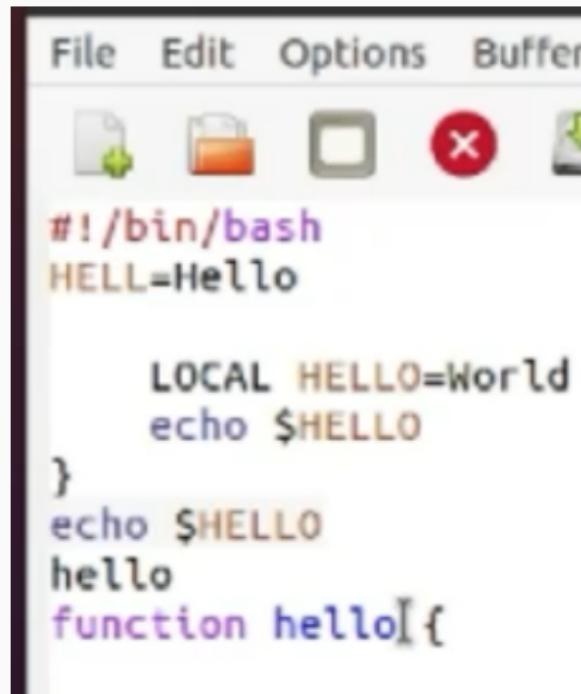
```
#!/bin/bash
HELLO=Hello

LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
function hello {
```

Figure 6: Вставка строки

## Выполнение лабораторной работы

5.3) Выделим область текста («Ctrl-space»)



```
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}

echo $HELLO
hello
function hello{
```

Figure 7: Выделение области текста

## Выполнение лабораторной работы

5.4) Скопируем область в буфер обмена («Alt-w»).

5.5) Вставим область в конец файла («Ctrl-y»)



The screenshot shows a terminal window with a menu bar containing File, Edit, Options, Buffers, and Terminal. Below the menu are several icons: a file with a plus sign, a folder, a square, a red circle with a white 'X', and a save icon. The main area displays a shell script:

```
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}

echo $HELLO
hello
function hello {
    echo $HELLO
```

A cursor is visible over the word "hello" in the final function definition.

Figure 8: Вставка области текста

## Выполнение лабораторной работы

5.6) Вновь выделим эту область(«Ctrl-space»)



```
#!/bin/bash
HELL=Hello

    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
function hello {
    echo $HELLO
```

Figure 9: Выделение области

## Выполнение лабораторной работы

5.7) Отменим последнее действие («Ctrl-/»)



```
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}

echo $HELLO
hello
function hello {
    echo $HELLO
```

Figure 11: Отмена последнего действия

## Выполнение лабораторной работы

- 6.1) Переместим курсор в начало строки («Ctrl-a»)



```
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}

echo $HELLO
hello
function hello{
    echo $HELLO
```

Figure 12: Перемещение курсора

## Выполнение лабораторной работы

6.2) Переместим курсор в конец строки («Ctrl-e»)



```
#!/bin/bash
HELL=Hello

    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO█
hello
function hello{
    echo $HELLO
```

Figure 13: Перемещение курсора

## Выполнение лабораторной работы

6.3) Переместим курсор в начало буфера («Alt-<»)



```
File Edit Options Buffers
! /bin/bash
HELL=Hello
LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
function hello {
echo $HELLO
```

Figure 14: Курсор в начале буфера

## Выполнение лабораторной работы

6.4) Переместим курсор в конец буфера («Alt->»)

```
#!/bin/bash
HELL=Hello

    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
function hello {
echo $HELLO█
```

Figure 15: Курсор в конце буфера

# Выполнение лабораторной работы

7.1) Выведем список активных буферов на экран («Ctrl-x»«Ctrl-b»)

The screenshot shows the Emacs interface with the following details:

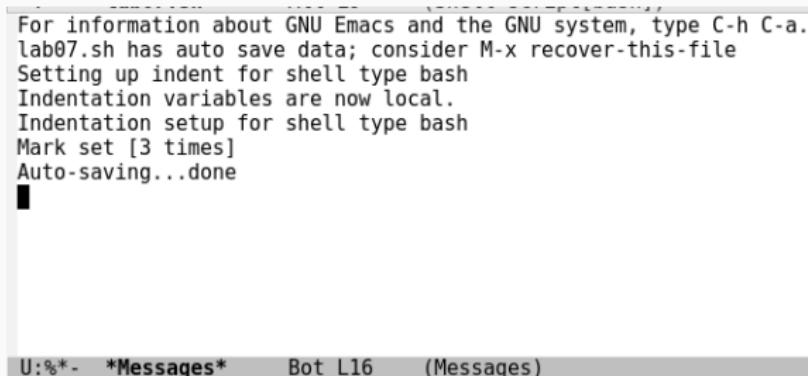
- Buffer List:** The main window displays the content of the buffer `lab10.sh`, which contains a shell script with variable assignments and function definitions.
- Minibuffer:** The bottom part of the screen shows the minibuffer with the text `U:%%- *Buffer List* All L1 (Buffer Menu)`.
- Buffer List Table:** A table at the bottom lists active buffers with their file names, modes, sizes, and types:

CRM Buffer	Size	Mode	File
lab10.sh	114	Shell-script[...]	~/lab10.sh
* GNU Emacs*	1006	Fundamental	
*scratch*	145	Lisp Interaction	
*Messages*	687	Messages	

Figure 16: Список активных буферов

## Выполнение лабораторной работы

7.2) Переместимся во вновь открытое окно («Ctrl-хо») со списком открытых буферов и переключимся на другой буфер (для этого необходимо нажать на «enter» после выбора необходимого буфера)



For information about GNU Emacs and the GNU system, type C-h C-a.  
lab07.sh has auto save data; consider M-x recover-this-file  
Setting up indent for shell type bash  
Indentation variables are now local.  
Indentation setup for shell type bash  
Mark set [3 times]  
Auto-saving...done

U:%\*- \*Messages\* Bot L16 (Messages)

Figure 17: Список активных буферов

## Выполнение лабораторной работы

7.3) Закроем это окно («Ctrl-x0»)



The screenshot shows a terminal window with a light gray background and a dark gray title bar. The title bar has five small colored icons: green, orange, blue, red, and gray. Below the title bar, there is a scroll bar on the left side of the window. The terminal contains the following text:

```
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO

echo $HELLO
hello
function hello {
    echo $HELLO
```

Figure 18: Закрытие окна

7.4) Теперь вновь переключимся между буферами, но уже без вывода их списка на экран («Ctrl-x b»)

## Выполнение лабораторной работы

8.1) Поделим фрейм на 4 части: разделим фрейм на два окна по вертикали («Ctrl-x 3»), а затем каждое из этих окон на две части по горизонтали («Ctrl-x 2»)

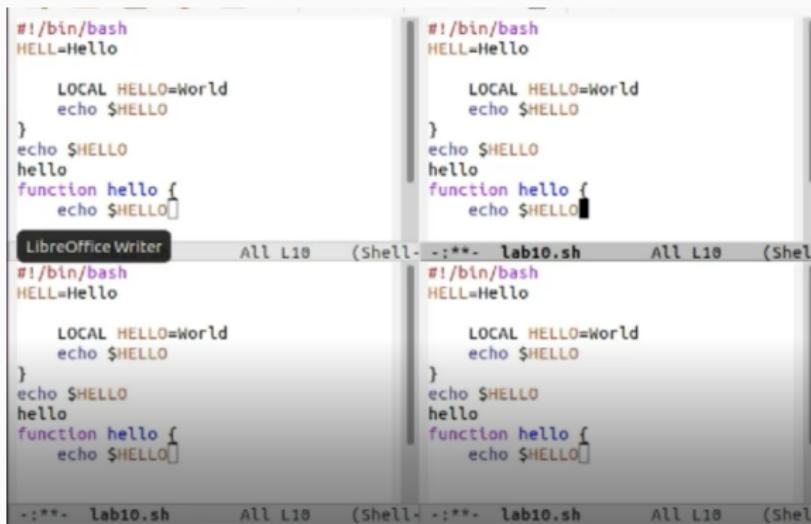


Figure 19: Переключение между буферами

## Выполнение лабораторной работы

8.2) В каждом из четырёх созданных окон откроем новый буфер (файл) и введем несколько строк текста. Для этого предварительно создадим эти файлы с помощью команд «touchexample1.txt», «touchexample2.txt», «touchexample3.txt», «touchexample4.txt»

```
nmbezruk@username: ~ touch example1.txt  
nmbezruk@username: ~ touch example2.txt  
nmbezruk@username: ~ touch example3.txt  
nmbezruk@username: ~ touch example4.txt  
nmbezruk@username: ~
```

Figure 20: Создание файлов



Figure 21: Вывод на экран

# Выполнение лабораторной работы

9.1) Переключимся в режим поиска («Ctrl-s») и найдем несколько слов, присутствующих в тексте

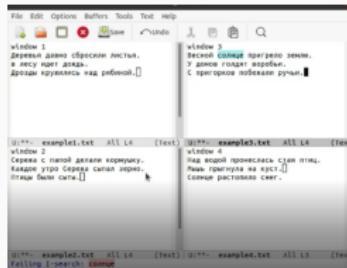


Figure 22: Поиск слова

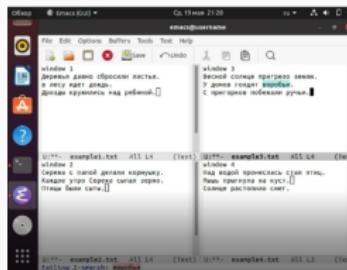


Figure 23: Поиск слова

# Выполнение лабораторной работы

9.2) Переключимся между результатами поиска, нажимая «Ctrl-s»

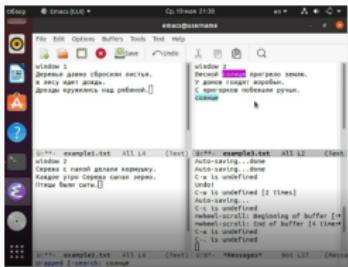


Figure 24: Переключение

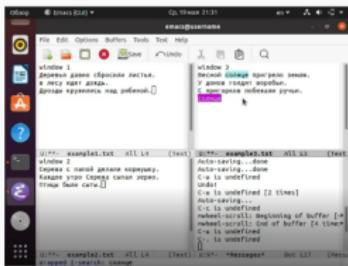


Figure 25: Переключение

# Выполнение лабораторной работы

## 9.3) Выйдем из режима поиска, нажав «Ctrl-g»

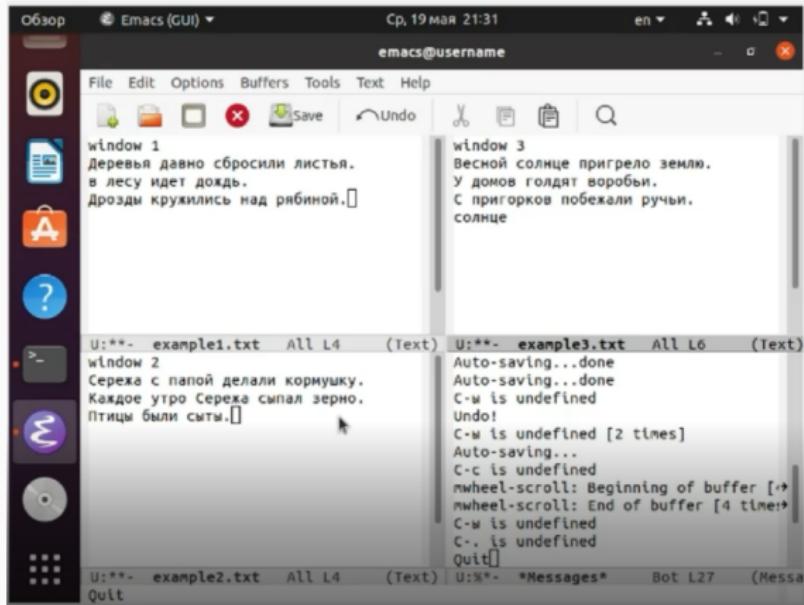


Figure 26: Переключение

## Выполнение лабораторной работы

9.4) Перейдем в режим поиска и замены («Alt-%»), введем текст, который следует найти и заменить, нажмем «enter», затем введем текст для замены. После того как будут подсвечены результаты поиска, нажмем «!» для подтверждения замены. Важно, чтобы курсор находился в начале текста.

The screenshot shows a text editor window with the following interface and content:

- Menu Bar:** File, Edit, Options, Buffers, Tools, Text, Help.
- Toolbar:** Icons for New, Open, Save, Undo.
- Text Area:** Shows two windows:
  - window 1:** Contains the text "деревья давно сбросили листья.  
в лесу идет дождь.  
дрозды кружились над рябиной."
  - window 2:** Contains the text "Сережа с папой делали кормушку.  
Каждое утро Сережасыпал зерно.  
Птицы были сыты.[]"
- Status Bar:** U:\*\*- example1.txt All L4 (Text)

# Выполнение лабораторной работы

The screenshot shows a text editor window with three visible panes:

- Window 1:** Displays a poem:

```
window 1
Деревья давно сбросили листья.
в лесу идет дождь.
Дрозды кружились над ивой.
```
- Window 2:** Displays another poem:

```
window 2
Сережа с папой делали кормушку.
Каждое утро Сережасыпал зерно.
Птицы были сыты.
```
- Status Bar:** Shows the file path and type for each window:
  - U:\*\*- example1.txt All L4 (Text)
  - U:\*\*- example2.txt All L4 (Text)
  - Replaced 1 occurrence

## Выполнение лабораторной работы

9.5) Пробуем другой режим поиска, нажав «Alt-so»  
Данный вид поиска отличается от обычного тем, что тут считывается строка поиска, которая трактуется как регулярное выражение, и не осуществляется поиск точного совпадения в тексте буфера. Регулярное выражение – это образец, который обозначает набор строк, возможно, и неограниченный набор.

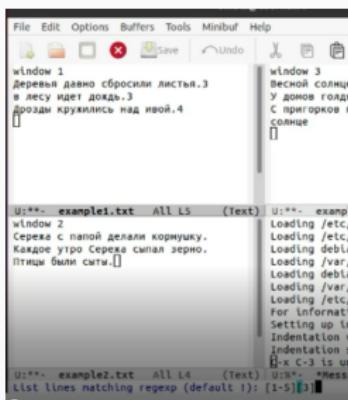


Figure 29: Замена

## Вывод

---

## Вывод

В ходе выполнения данной лабораторной работы я познакомилась с операционной системой Linux и получила практические навыки работы с редактором Emacs