

# Лабораторная работа № 11. Текстовый редактор emacs

Отчёт

---

Сергеев Д. О.

26 апреля 2025

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Сергеев Даниил Олегович
- Студент
- Направление: Прикладная информатика
- Российский университет дружбы народов
- 1132246837@pfur.ru

## 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.

## Ход выполнения лабораторной работы

---



## Выполнение упражнений с основными командами emacs

Откроем emacs в фоновом режиме. Перейдем в каталог lab07 и создадим файл lab07.sh с помощью комбинации C-x C-f в самом emacs.

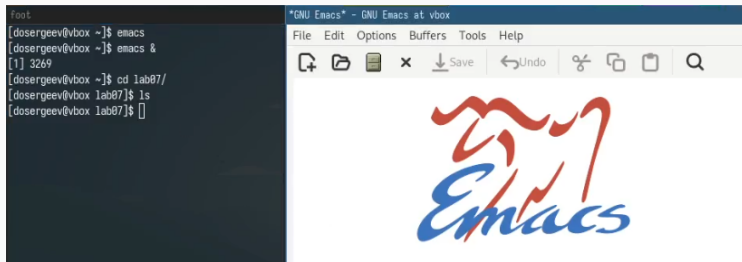
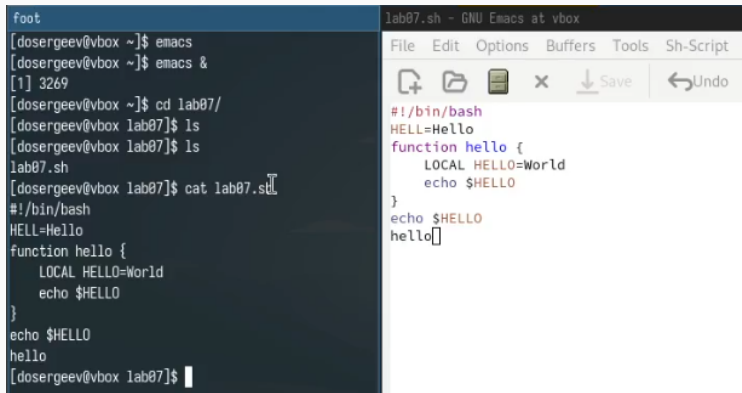


Рис. 1: Открытый emacs.

## Выполнение упражнений с основными командами emacs

Наберем в файле указанный текст и сохраним его с помощью комбинации клавиш C-x C-s.



The image shows two side-by-side windows. The left window is a terminal with a dark background, showing a series of commands and their outputs. The right window is the GNU Emacs editor, displaying a shell script with syntax highlighting. The terminal commands include opening Emacs, running it in the background, changing to the 'lab07' directory, listing files, and using 'cat' to view the contents of 'lab07.sh'. The Emacs window shows the same script content: a shebang, a variable assignment, a function definition, and two echo statements.

```
foot
[dosergeev@vbox ~]$ emacs
[dosergeev@vbox ~]$ emacs &
[1] 3269
[dosergeev@vbox ~]$ cd lab07/
[dosergeev@vbox lab07]$ ls
[dosergeev@vbox lab07]$ ls
lab07.sh
[dosergeev@vbox lab07]$ cat lab07.sh
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
[dosergeev@vbox lab07]$
```

```
lab07.sh - GNU Emacs at vbox
File Edit Options Buffers Tools Sh-Script
[Icons] Save Undo
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
```

Рис. 2: Сохранение кода.

Проделаем с текстом несколько примеров. Стандартные операции редактирования с помощью комбинаций клавиш:

1. Вырежем 7 строку (echo \$HELLO) командой C-k.
2. Вставим её в конец файла с помощью C-y.
3. Выделим в область слово hello в 7 строке, предварительно перейдя в режим выделения командой C-space.
4. Скопируем его в буффер с помощью RAlt(M)-w.
5. Вставим выделенную область в конец файла.
6. Вновь выделим слово hello и вырежем его командой C-w.
7. Отменим последнее действие с помощью C-/\_.

Далее используем команды по перемещению курсора. Переместим курсор в:

1. начало строки с помощью C-a.
2. конец строки с помощью C-e.
3. начало буфера (файла) с помощью M-<.
4. конец буфера (файла) с помощью M->.

Теперь попробуем управлять буферами и окнами:

1. Выведем список активных буферов на экран командами C-x C-b.
2. Переместимся в открытое окно командой C-x o. Переключимся на другой буфер, нажав сочетание клавиш C-x b и выбрав окно Messages.
3. Закроем окно списка активных буферов с помощью C-x 0.
4. Переключимся из буфера файла lab07.sh на любой другой файл командой C-x b.
5. Поделим рабочую область на 4 части: сначала пополам по вертикали (C-x 3), а потом по горизонтали, выбрав каждое окно по отдельности (C-x 2).
6. Перейдем в каждое окно командой C-x o, откроем или создадим новые файлы с помощью C-x C-f и введем в них текст.

## Выполнение упражнений с основными командами emacs

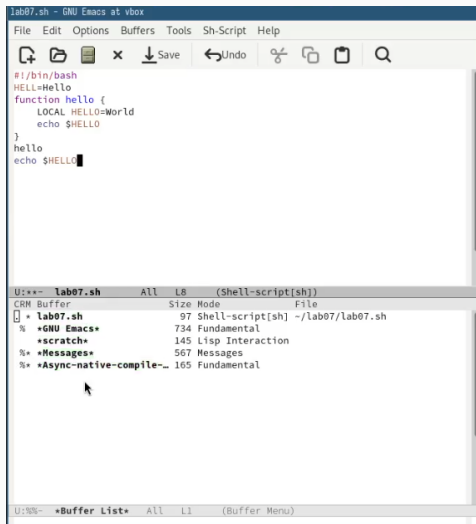
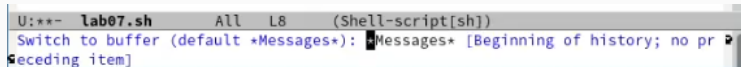


Рис. 3: Окно со списком буфферов.



The screenshot shows an Emacs window with a title bar that reads "U:\*\*- lab07.sh All L8 (Shell-script[sh])". The main text area displays the command "Switch to buffer (default \*Messages\*):" followed by a cursor, then "\*Messages\*" and "[Beginning of history; no preceding item]".

Рис. 4: Переключение буфферов.

# Выполнение упражнений с основными командами emacs

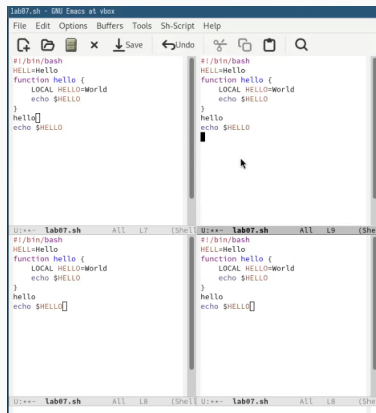


Рис. 5: Разделение на 4 окна.

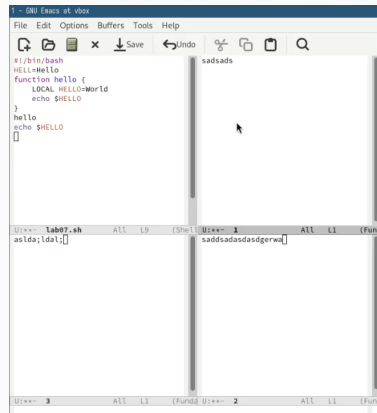
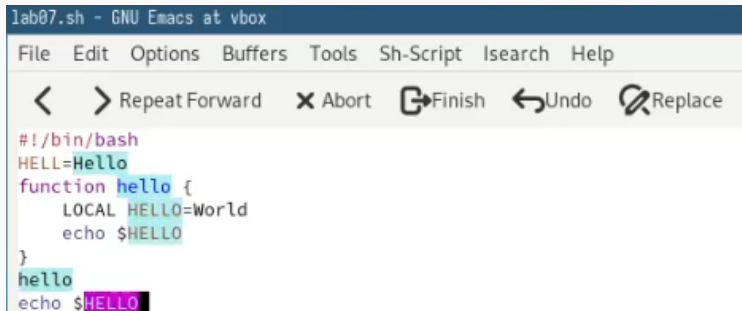


Рис. 6: Открытие новых файлов



## Выполнение упражнений с основными командами emacs

Поработаем с режимом поиска. Переключимся в него с помощью команды C-s. Попробуем найти слово hello, с помощью той же команды переключимся между результатами поиска (выделено розовым цветом). Выйдем из режима с помощью комбинации C-g.

The image shows a terminal window titled 'lab07.sh - GNU Emacs at vbox'. The Emacs interface includes a menu bar with 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', 'Isearch', and 'Help'. Below the menu is a toolbar with icons for navigation and search: left arrow, right arrow, 'Repeat Forward', 'Abort' (with an 'X' icon), 'Finish' (with a 'G' icon), 'Undo' (with a left arrow icon), and 'Replace' (with a pencil icon). The main text area contains a shell script:

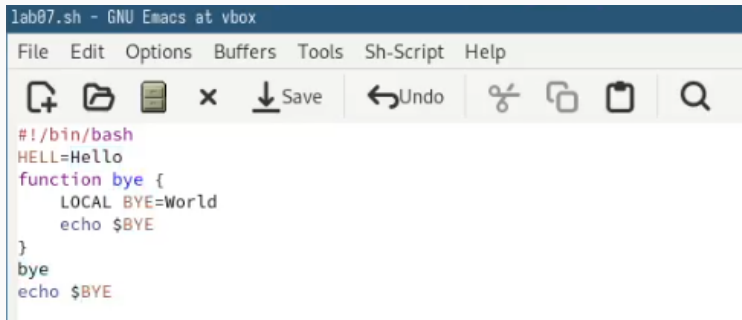
```
#!/bin/bash
HELL=Hello
function hello {
  LOCAL HELLO=World
  echo $HELLO
}
hello
echo $HELLO
```

The word 'hello' on the fourth line is highlighted in pink. The word 'HELLO' on the last line is highlighted in pink. The cursor is at the end of the last line.

Рис. 7: Обычный режим поиска

## Выполнение упражнений с основными командами emacs

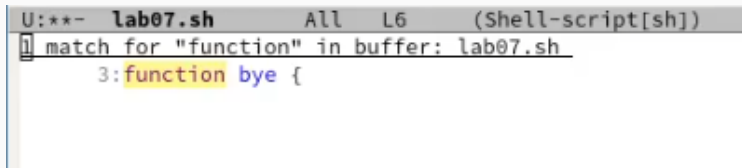
Перейдем в режим поиска и замены. Заменяем слова hello на bye. Для этого используем команду M-%, напишем hello и нажмем Enter, затем введем bye. После этого пропишем ! для подтверждения замены.

The image shows a screenshot of the GNU Emacs editor window titled 'lab07.sh - GNU Emacs at vbox'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The toolbar contains icons for opening a file, saving, undo, redo, and search. The main text area displays a shell script with the following content:

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

Рис. 8: Замена слова hello на bye

Наконец испробуем другой режим поиска, нажав M-s o. В отличие от обычного режима, открывается отдельное окно со списком всех строк файла, содержащих указанное слово.



```
U:***- lab07.sh All L6 (Shell-script[sh])
1 match for "function" in buffer: lab07.sh
  3: function bye {
```

Рис. 9: Окно альтернативного поиска

1. Emacs - экранный редактор текста, написанный на языке высокого уровня Elisp.
2. Нестандартные комбинации клавиш и отсутствие управляющих визуальных кнопок могут сделать его сложным для освоения новичком.
3.
  - Буфер - это часть окна, в которой представлено изменение текста открытого файла (временный объект, а не сам файл).
  - Окно - это область программы, содержащая один или несколько буферов.

4. Да, можно.

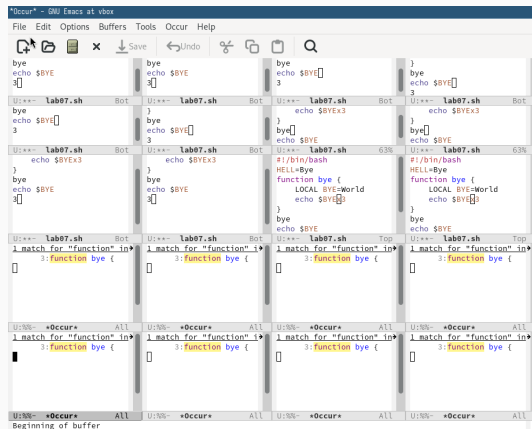


Рис. 10: 16 буферов в одном окне.

5. По умолчанию при запуске emacs открывается окно scratch.

6.

- C-c | : Ctrl+c Shift+
- C-c C-| : Ctrl+c Ctrl+Shift+

7. Чтобы поделить текущее окно на две части нужно нажать сочетание клавиш:

- по вертикали C-x 3
- по горизонтали C-x 2

8. Файлы настроек emacs хранятся в каталоге `~/emacs.d/`.
9. Клавиша Backspace выполняет функцию удаления символов в буфере. Её можно переназначить, добавив в файл конфигурации `~/emacs.d/init.el` строку:
  - к примеру, для глобального переназначения: `(global-set-key (kbd "<backspace>") 'функция)`, стандартная функция для backspace - `delete-char`.
10. На первый взгляд Vim, в отличие от Emacs, более удобный. Он имеет более привычные комбинации, быстрый и понятный. Emacs в свою очередь тяжелее для освоения и требует настройки для удобного использования.

## Вывод

---



В результате выполнения лабораторной работы я получил практические навыки работы с редактором Emacs и освоил комбинации клавиш для его удобного использования.