

Лабораторная работа №1

Операционные системы

Павлова Т. Ю.

Российский университет дружбы народов, Москва, Россия

Целью данной лабораторной работы является ознакомление с операционной системой Linux, а также получение практические навыков работы с редактором Emacs.

1. Открыть emacs.
2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).
3. Наберите текст
4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).
5. Прodelать с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш. 5.1. Вырезать одной командой целую строку (C-k). 5.2. Вставить эту строку в конец файла (C-y). 5.3. Выделить область текста (C-space). 5.4. Скопировать область в буфер обмена (M-w). 5.5. Вставить область в конец файла. 5.6. Вновь выделить эту область и на этот раз вырезать её (C-w). 5.7. Отмените последнее действие (C-/).
6. Научитесь использовать команды по перемещению курсора. 6.1. Переместите курсор в начало строки (C-a). 6.2. Переместите курсор в конец строки (C-e). 6.3. Переместите курсор в начало буфера (M-<). 6.4. Переместите курсор в конец буфера (M->).
7. Управление буферами. 7.1. Вывести список активных буферов на экран (C-x C-b). 7.2.

Основные термины Emacs

Определение 1 Буфер — объект, представляющий какой-либо текст. Буфер может содержать что угодно, например, результаты компиляции программы или встроенные подсказки. Практически всё взаимодействие с пользователем, в том числе интерактивное, происходит посредством буферов.

Определение 2 Фрейм соответствует окну в обычном понимании этого слова. Каждый фрейм содержит область вывода и одно или несколько окон Emacs.

Определение 3 Окно — прямоугольная область фрейма, отображающая один из буферов. Каждое окно имеет свою строку состояния, в которой выводится следующая информация: название буфера, его основной режим, изменялся ли текст буфера и как далеко вниз по буферу расположен курсор. Каждый буфер находится только в одном из возможных основных режимов. Существующие основные режимы включают режим Fundamental (наименее специализированный), режим Text, режим Lisp, режим C, режим Texinfo и другие. Под второстепенными режимами понимается список режимов, которые включены в данный момент в буфере выбранного окна.

Определение 4 Область вывода — одна или несколько строк внизу фрейма, в которой Emacs выводит

Выполнение лабораторной работы

Откройте emacs (рис. 1).

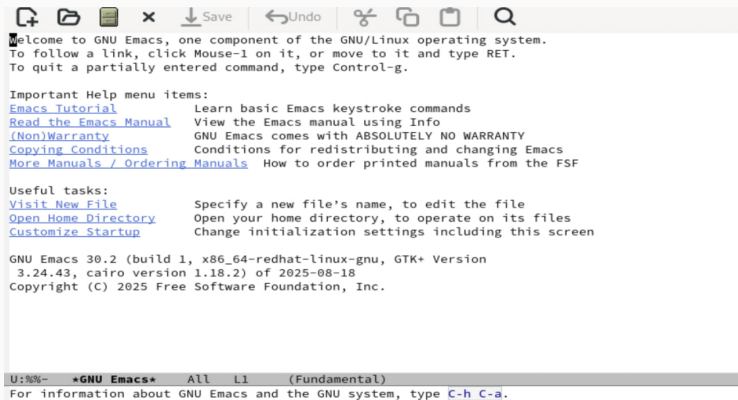
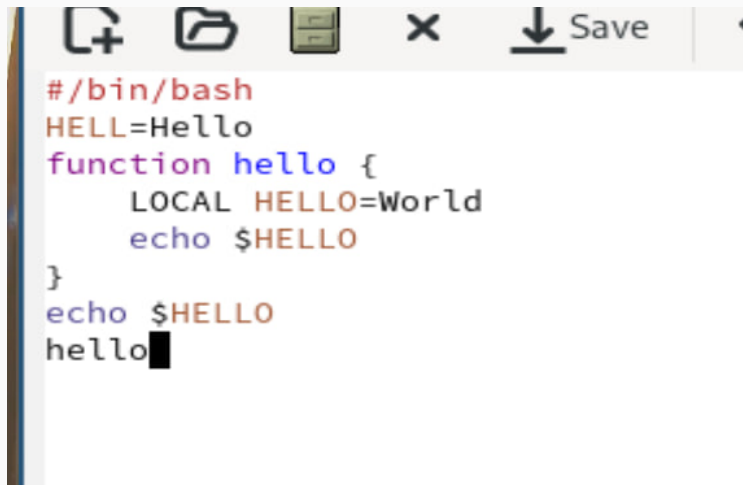


Рис. 1: Открытый emacs

Созданный и заполненный файл

Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f). Наберите текст (рис. 2).

A screenshot of a text editor window. The title bar at the top shows icons for undo, redo, save, close, and a 'Save' button. The editor area contains the following shell script code:

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

The text is color-coded: red for the shebang, orange for variable names, purple for function names, and black for other text. A black cursor is positioned at the end of the last line, 'hello'.

Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s) (рис. 3).

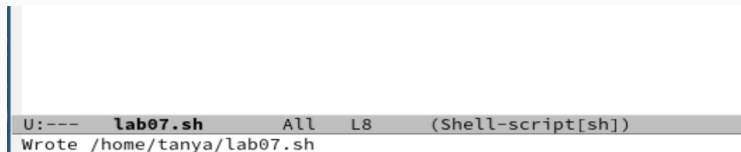
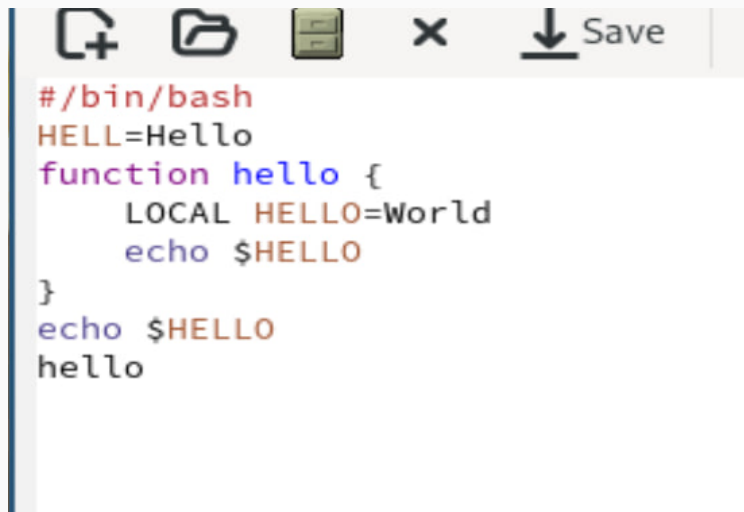


Рис. 3: Сохранение файла

Управление буферами. 7.1. Вывести список активных буферов на экран (C-x C-b). 7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых буферов и переключитесь на другой буфер. 7.3. Закройте это окно (C-x 0). 7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b). Прodelать с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш. 5.1. Вырезать одной командой целую строку (C-k). 5.2. Вставить эту строку в конец файла (C-y). 5.3. Выделить область текста (C-space). 5.4. Скопировать область в буфер обмена (M-w). 5.5. Вставить область в конец файла. 5.6. Вновь выделить эту область и на этот раз вырезать её (C-w). 5.7. Отмените последнее действие (C-/). (рис. 4), (рис. 5), (рис. 6), (рис. 7), (рис. 8), (рис. 9).




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



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

Рис. 5: Вставленная строка



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

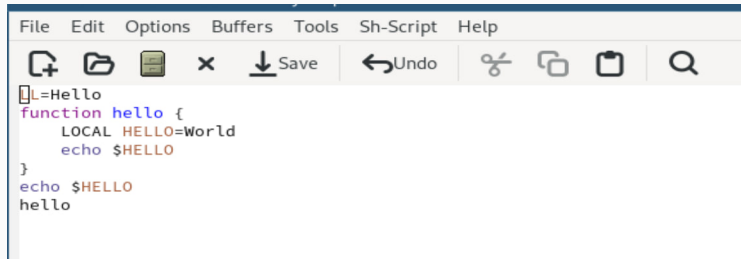
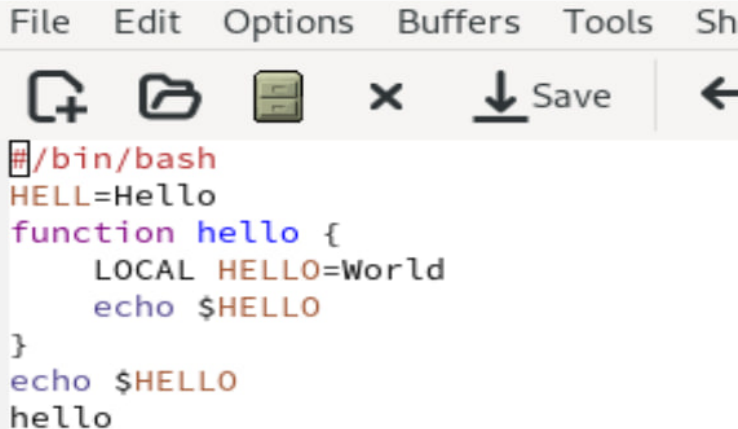


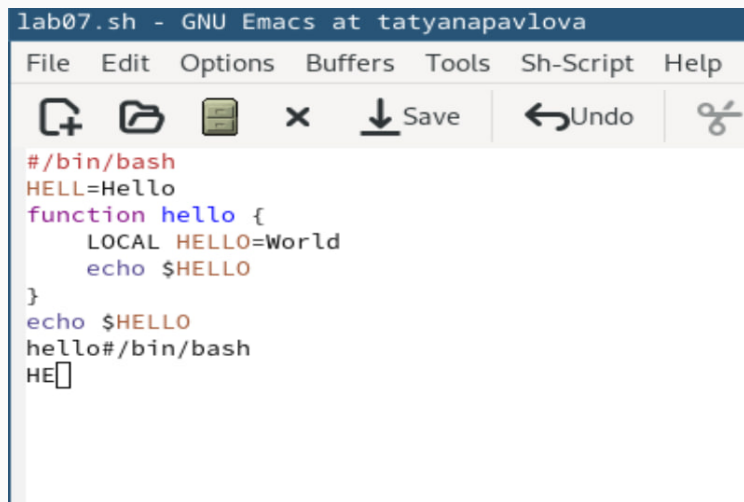
Рис. 7: Вырезанная область текста



The image shows a code editor window with a menu bar and a toolbar. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', and 'Sh'. The toolbar contains icons for opening a file, saving a file, closing a file, and a 'Save' button with a download icon. The code editor displays a shell script with the following content:

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

Скопированная и вставленная в конец область

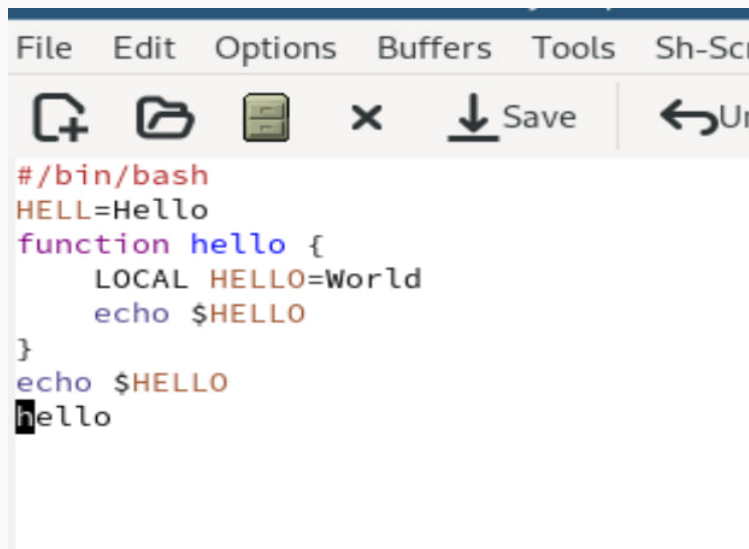


```
lab07.sh - GNU Emacs at tatyana.pavlova
File Edit Options Buffers Tools Sh-Script Help
[Icons: Copy, Paste, Save, Undo, etc.]

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

Рис. 9: Скопированная и вставленная в конец область

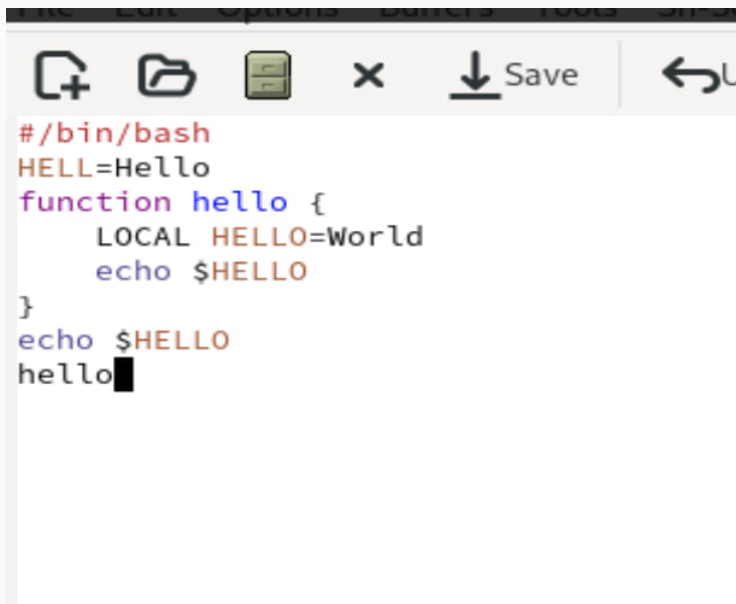
Научитесь использовать команды по перемещению курсора. 6.1. Переместите курсор в начало строки (C-a). 6.2. Переместите курсор в конец строки (C-e). 6.3. Переместите курсор в начало буфера (M-<). 6.4. Переместите курсор в конец буфера (M->) (рис. 10), (рис. 11).



The image shows a terminal window with a menu bar (File, Edit, Options, Buffers, Tools, Sh-Sc) and a toolbar with icons for copy, paste, save, and other functions. The terminal content is a Bash script:

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

A black cursor is positioned at the beginning of the last line, "hello".



The screenshot shows a code editor window with a menu bar at the top containing 'File', 'Edit', 'Options', 'Buffers', 'Tools', and 'Shell'. Below the menu bar is a toolbar with icons for opening a file, saving a file, closing a file, and a 'Save' button. The main area of the editor contains a shell script. The script defines a function 'hello' that prints 'World' and then prints the value of 'HELLO' (which is 'Hello'). The cursor is positioned at the end of the last line of the script, 'hello'.

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

Управление буферами. 7.1. Вывести список активных буферов на экран (С-х С-b). 7.2. Переместитесь во вновь открытое окно (С-х) о со списком открытых буферов и переключитесь на другой буфер. 7.3. Закройте это окно (С-х 0). 7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (С-х b) (рис. 12), (рис. 13).

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

U:***- lab07.sh		All	L8	(Shell-script[sh])
CRM	Buffer	Size	Mode	File
[]	* lab07.sh	96	Shell-script[sh]	~/lab07.sh
%	*GNU Emacs*	904	Fundamental	
	scratch	147	Lisp Interaction	
%*	*Messages*	1651	Messages	

Рис. 12: Список буферов

Переключение на другой буфер

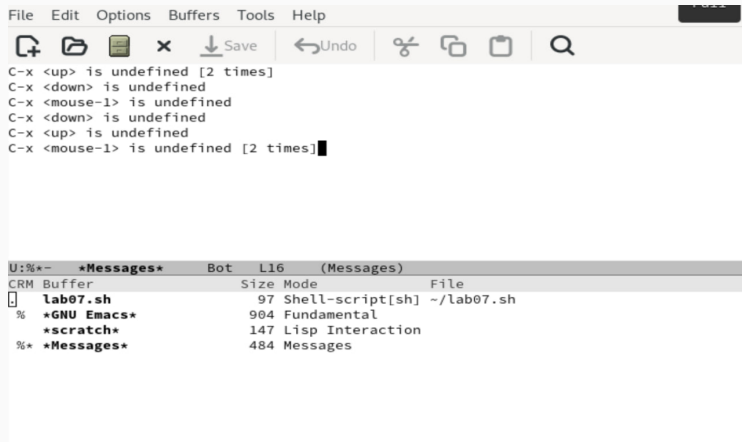
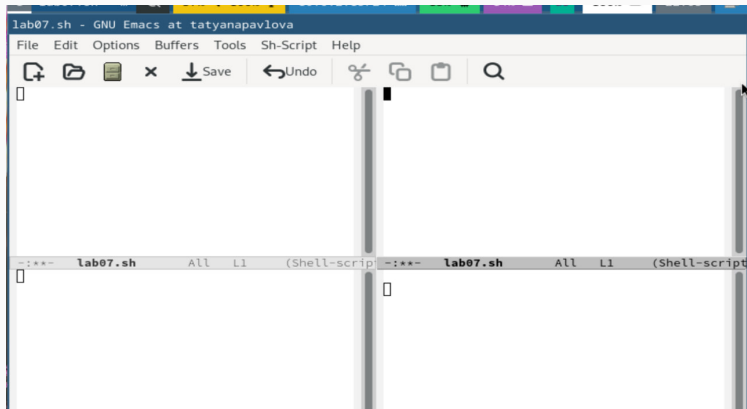
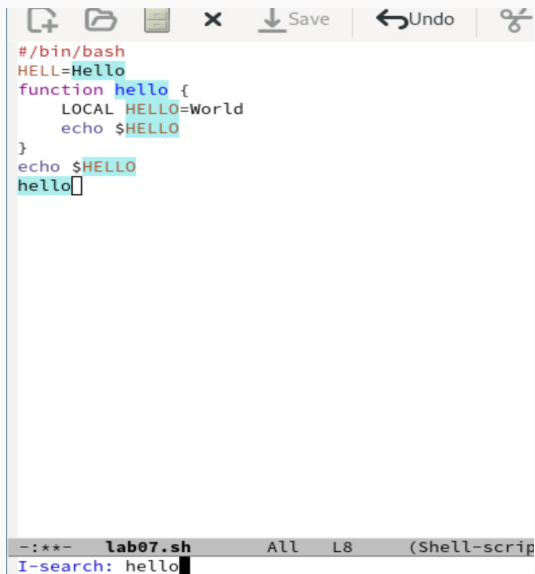


Рис. 13: Переключение на другой буфер

Управление окнами. 8.1. Поделите фрейм на 4 части: разделите фрейм на два окна по вертикали (С-х 3), а затем каждое из этих окон на две части по горизонтали (С-х 2) 8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста (рис. 14).



Режим поиска 9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте. 9.2. Переключайтесь между результатами поиска, нажимая C-s. 9.3. Выйдите из режима поиска, нажав C-g. 9.4. Перейдите в режим поиска и замены (M-%), введите текст, который следует найти и заменить, нажмите Enter, затем введите текст для замены. После того как будут подсвечены результаты поиска, нажмите ! для подтверждения замены. 9.5. Испробуйте другой режим поиска, нажав M-s o (рис. 15), (рис. 16).



The image shows a code editor window with a light gray toolbar at the top containing icons for file operations (copy, paste, save, close) and editing (undo, redo). The main area displays a shell script with syntax highlighting: `#!/bin/bash` (red), `HELL=Hello` (green), `function hello {` (purple), `LOCAL HELLO=World` (green), `echo $HELLO` (green), `}` (purple), `echo $HELLO` (green), and `hello` (green) followed by a cursor. At the bottom, a search bar is active, showing `I-search: hello` with a cursor. The status bar at the very bottom indicates the file is `lab07.sh`, line `L8`, and is a `(Shell-script)`.

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

-:***- lab07.sh All L8 (Shell-script)

I-search: hello

< > Repeat Forward ✕ Abort

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

При выполнении данной лабораторной работы, я ознакомилась с операционной системой Linux, а также получила практические навыки работы с редактором Emacs.