

Лабораторная работа №9: Презентация.

Текстовый редактор emacs.

Евдокимов Максим Михайлович. Группа - НФИбд-01-20.¹

27 декабря, 2023, Москва, Россия

¹Российский Университет Дружбы Народов

Цели и задачи работы

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

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

Указание к работе

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 различные сообщения, а также запрашивает подтверждения и дополнительную информацию от пользователя.

Определение 5. Минибуфер используется для ввода дополнительной информации и всегда отображается в области вывода.

Определение 6. Точка вставки — место вставки (удаления) данных в буфере.

Процесс выполнения лабораторной работы

Процесс выполнения лабораторной работы

1. Открыть emacs.

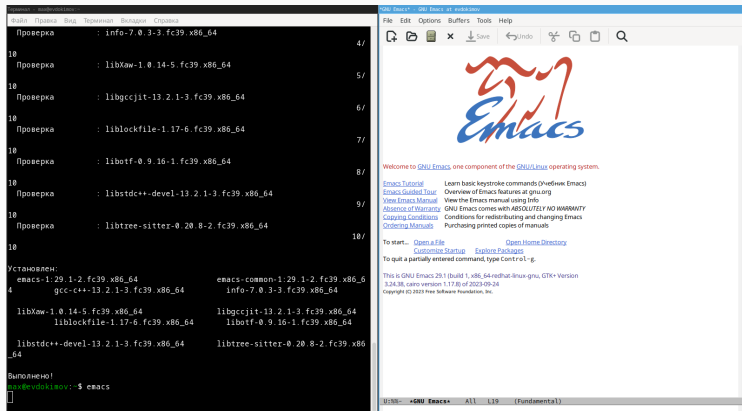


Рис. 1: Открытие emacs

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

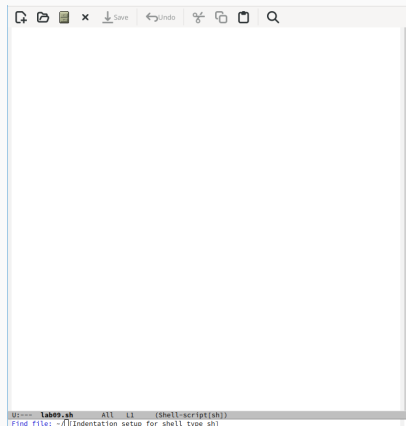
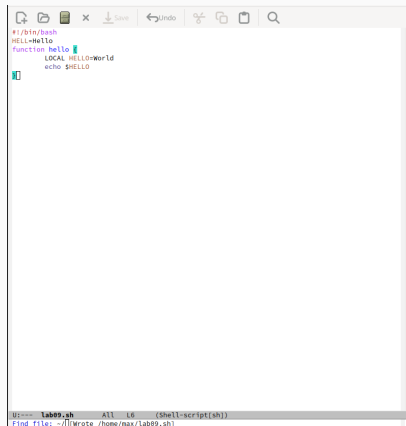


Рис. 2: Создание файла

3. Наберите текст указанный в задание.



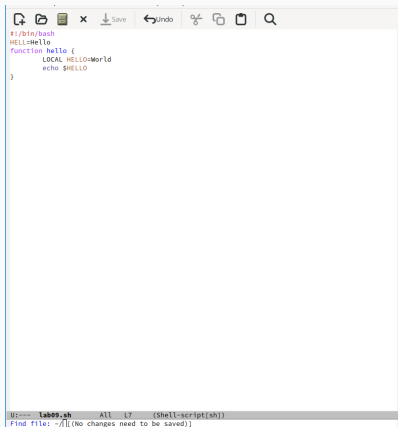
The image shows a code editor window with a light gray background. The top toolbar contains icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search. The main text area contains a shell script with the following content:

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

The script is written in a syntax-highlighted format: the shebang is red, the variable assignment is blue, the function definition is purple, and the function body is black. A green cursor is positioned at the end of the first line of the function body. The status bar at the bottom of the editor displays the file name 'lab09.sh', the line and column number 'All 16', the file type '(shell-script(sh))', and the file path 'Find file: ~/Wrote /home/max/lab09.sh'.

Рис. 3: Ввод текста

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

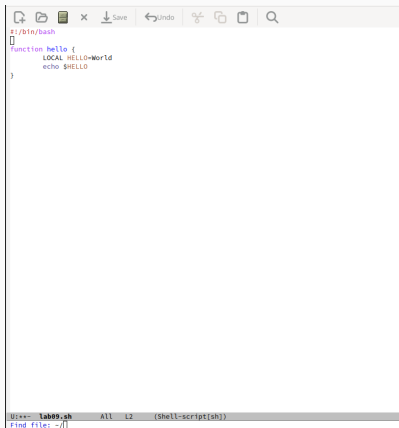


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

U:--- lab09.sh  All  L7  (Shell-script(sh))
Find file: ~/1 (No changes need to be saved)
```

Рис. 4: Сохранение файла

5. Прodelать с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш:



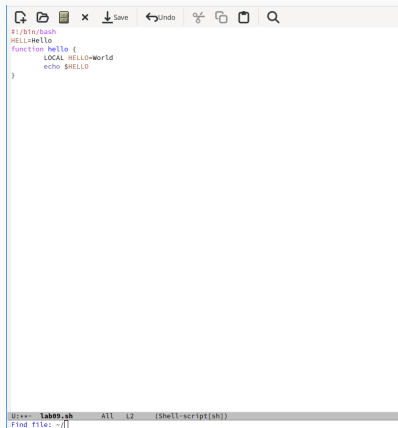
The image shows a screenshot of a text editor window. The title bar at the top contains icons for file operations (open, save, close) and editing (undo, redo, cut, copy, paste, search). The main text area contains a shell script snippet:

```
#!/bin/bash
function hello {
    local HELLO=World
    echo $HELLO
}
```

At the bottom of the window, there is a status bar with the text: `Ucs+ Lab09.sh All L2 (Shell-script[sh])`. Below the status bar, there is a prompt `Find file: ~/` followed by a cursor.

Рис. 5: Редактировать текст

5.1. Вырезать одной командой целую строку (C-k).

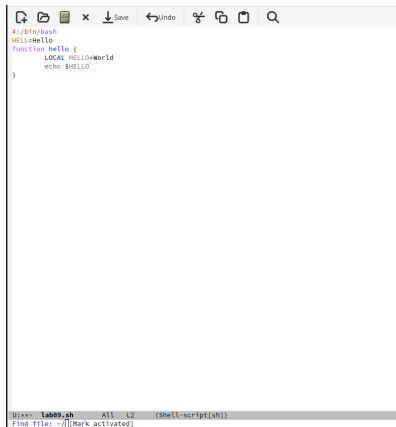


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

U:***- lab09.sh All L2 (shell-script[sh])
Find file: ~/[]

Рис. 6: Ведение строк

5.2. Вставить эту строку в конец файла (С-у).



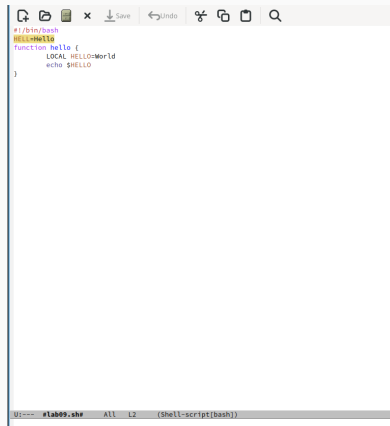
The screenshot shows a code editor window with a light gray background. The top toolbar contains icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search. The code is written in a shell script format with syntax highlighting: comments are purple, variable assignments are blue, function definitions are green, and function bodies are black. The script defines a function named 'hello' that sets a local variable 'HELLO' to 'World' and prints it. The status bar at the bottom indicates the current file is 'lab09.sh' and the editor is in 'shell-script(sh)' mode.

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

U:*** lab09.sh All L2 (shell-script(sh))
Find file: ~/[] [Mark activated]

Рис. 7: Вставка текста в конец

5.3. Выделить область текста (C-space).

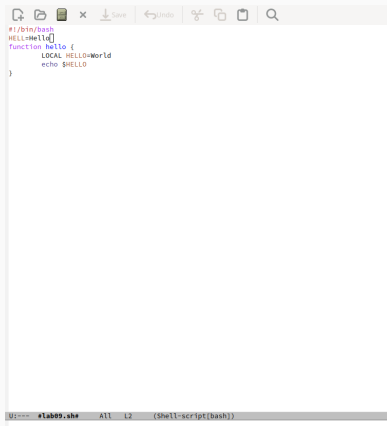


```
#!/bin/bash
HELLO=Hello
function hello {
    local HELLO=World
    echo $HELLO
}
```

U:--- #lab09.shv All L2 (Shell-script(bash))

Рис. 8: Выделение области

5.4. Скопировать область в буфер обмена (M-w).

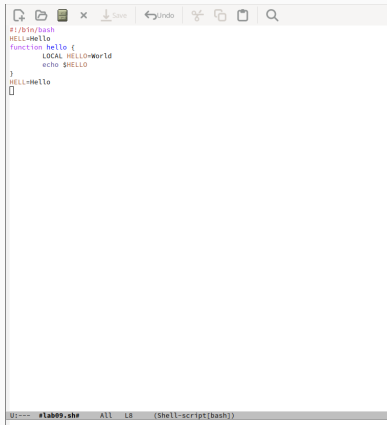


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL_HELL=World
    echo $HELL
}
```

U:~ # Lab09.sh All L2 (Shell-script(bash))

Рис. 9: Копирование в буфер обмена

5.5. Вставить область в конец файла.

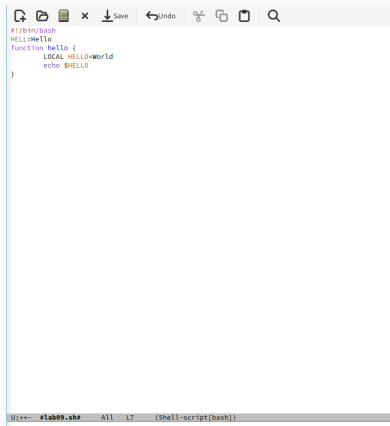


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
HELL=Hello
```

The screenshot shows a code editor window with a toolbar at the top containing icons for file operations (open, save, close, download), editing (undo, redo, cut, copy, paste), and search. The main area displays a shell script. The script defines a function named 'hello' that takes a local variable 'HELLO' and prints its value. A variable 'HELL' is assigned the value 'Hello'. The status bar at the bottom indicates the file is 'elab09.sh', line 18, and it's a shell script using bash.

Рис. 10: Вставка из буфера обмена

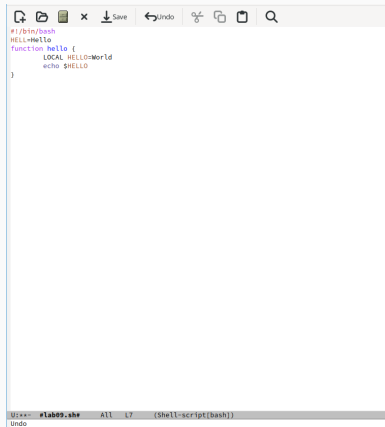
5.6. Вновь выделить эту область и на этот раз вырезать её (C-w).



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

Рис. 11: Вырезание области в буфер обмена

5.7. Отмените последнее действие (C-/).



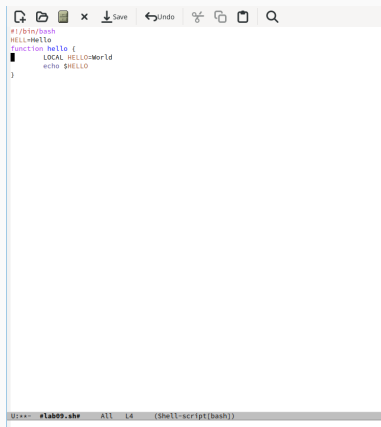
```
#!/bin/bash
HELLO=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

U:xxx- #lab09.shv All L7 (shell-script(bash))
Undo

Рис. 12: Отмена действий

6. Научитесь использовать команды по перемещению курсора:

6.1. Переместите курсор в начало строки (C-a).



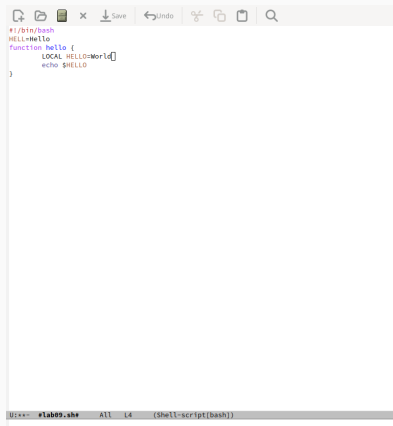
The screenshot shows a code editor window with a light gray background. At the top is a toolbar with icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search. Below the toolbar, the code is as follows:

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

A blue vertical line indicates the current cursor position, which is at the beginning of the line containing the function definition. The status bar at the bottom of the editor displays the text: "U:== #lab09.sh All 14 (Shell-script(bash))".

Рис. 13: Переход в начало строки

6.2. Переместите курсор в конец строки (C-e).



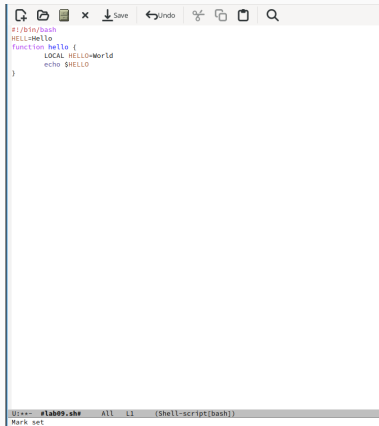
The screenshot shows a terminal window with a light gray title bar containing icons for file operations (open, save, delete, copy, paste, search) and a magnifying glass. The terminal content is as follows:

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

The cursor is positioned at the end of the closing brace of the function definition on the fourth line. At the bottom of the terminal, a status bar displays the text: `U:~# #lab09.sh# All L4 (Shell-script(bash))`.

Рис. 14: Переход в конец строки

6.3. Переместите курсор в начало буфера (M-<).

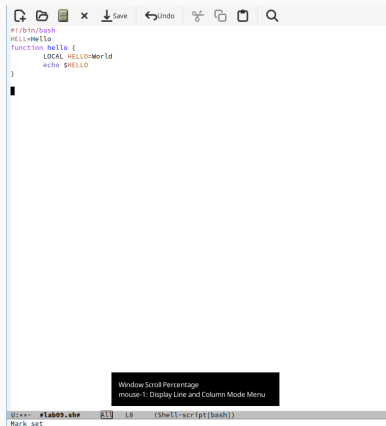


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

The screenshot shows a terminal window with a light gray title bar containing icons for file operations and search. The terminal content displays a shell script with a function named 'hello'. The cursor is positioned at the beginning of the first line of the script. The terminal's status bar at the bottom indicates the user is 'Uz***' in a window titled '#lab09.sh'.

Рис. 15: Перещение в начало буфера курсора

6.4. Переместите курсор в конец буфера (M->).



The screenshot shows a terminal window with a light gray background. At the top is a toolbar with icons for file operations (open, save, close, download), editing (undo, redo, cut, copy, paste), and search. The terminal content is a bash script:

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

```

The cursor is positioned at the end of the last line. A black tooltip box is visible over the terminal area, containing the text:

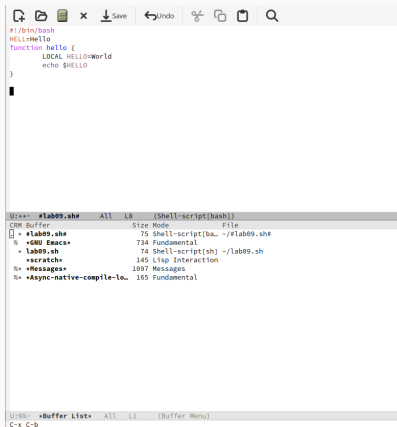
```
Window Scroll Percentage
mouse-1: Display Line and Column Mode Menu
```

At the bottom of the terminal, a status bar shows the prompt 'U:xxx@lab09.shv', a file icon, the filename 'l8', and the shell type '(shell-script(bash))'. Below the status bar, the text 'Mark set' is visible.

Рис. 16: Перещение в конец буфера курсора

7. Управление буферами:

7.1. Вывести список активных буферов на экран (C-x C-b).



The screenshot shows the Emacs editor interface. The main window displays a shell script with the following content:

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

Below the main window, the buffer list is displayed, showing the following buffers:

Buffer	Size	Mode	File
lab09.sh	75	Shell-script[ba...	~/lab09.sh
GNU Emacs	734	Fundamental	
*lab09.sh	74	Shell-script[sh]	~/lab09.sh
scratch	145	Lisp Interaction	
Messages	1897	Messages	
*Async-native-compile-lo...	165	Fundamental	

The status bar at the bottom indicates the current buffer is *lab09.sh* and the command C-x C-b was used to display the buffer list.

Рис. 17: Список активных буферов

7.2. Переместитесь во вновь открытое окно (С-х) о со списком открытых буферов и переключитесь на другой буфер.

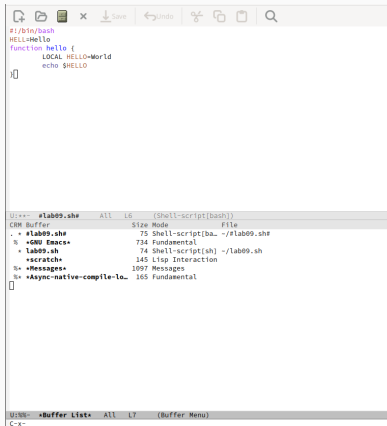
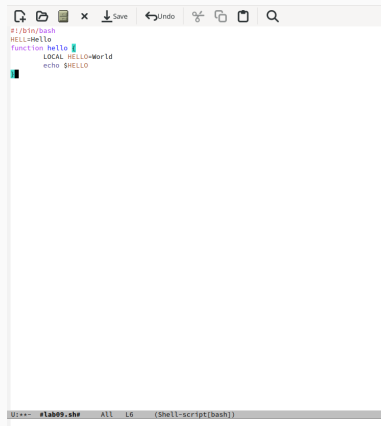


Рис. 18: Перемещение между буферами

7.3. Закройте это окно (С-х 0).



The image shows a terminal window with a light gray title bar and a toolbar. The toolbar contains icons for file operations (open, save, close, print) and editing (undo, redo, copy, paste, search). The terminal content shows a shell script being executed. The prompt is `#!/bin/bash`. The first line of the script is `HELL=Hello`. The second line is `function hello`. The third line is `LOCAL HELLO=World`. The fourth line is `echo $HELLO`. The output of the script is `HELLO`. The terminal window has a status bar at the bottom that reads `U:*** #lab09.sh All L6 (Shell-script{bash})`.

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

U:*** #lab09.sh All L6 (Shell-script{bash})

Рис. 19: Заккрытие сторонних окон

7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b).

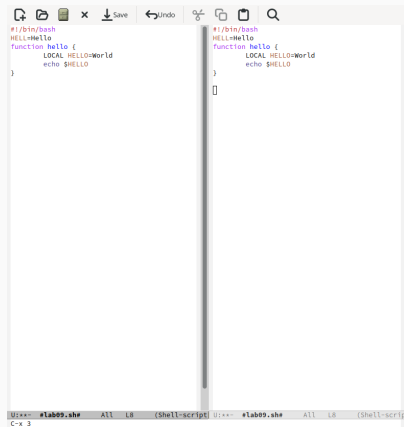


Рис. 20: Переключение между буферами

8. 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) (см. рис. 9.1).

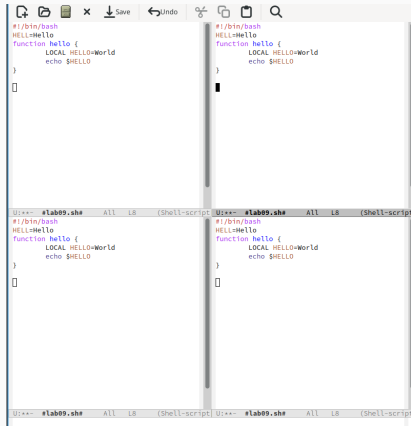


Рис. 21: Разделение на 2 буфера

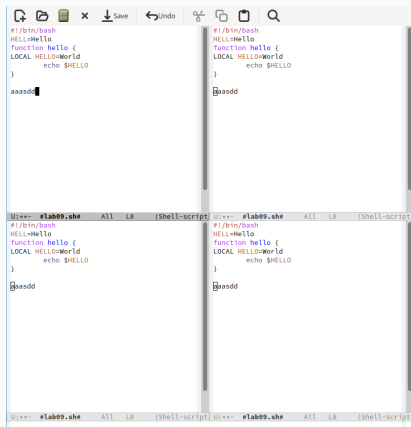
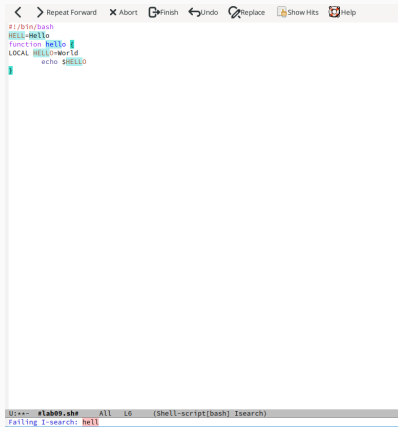


Рис. 22: Разделение на 4 буфера

8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста.



The screenshot shows a Vim editor window with a menu bar at the top containing: < > Repeat Forward, x Abort, G Finish, ↶ Undo, ↷ Replace, Show Hits, and Help. The main text area contains a shell script:

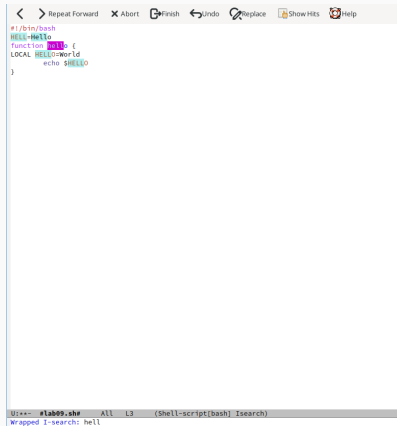
```
#!/bin/bash
HELL=Hello
function hello
LOCAL HELLO=World
    echo $HELLO
```

The cursor is at the end of the script. At the bottom, a status bar shows: U:-- #lab09.sh All L6 (Shell-script[bash] Isearch). Below the status bar, a search window displays: Failing I-search: hell.

Рис. 23: Одновременное редактирование нескольких файлов

9. Режим поиска:

9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте.



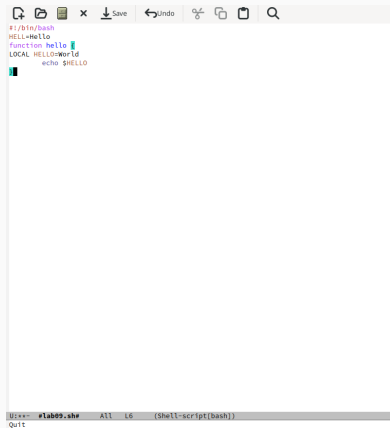
The screenshot shows a terminal window with a menu bar at the top containing icons for navigation and editing: back, forward, Repeat Forward, Abort, Finish, Undo, Replace, Show Hits, and Help. The main area displays the contents of a file named 'lab09.shr' in a syntax-highlighted format. The script starts with a shebang line '#!/bin/bash', followed by a line 'HELL=Hello'. Then, a function 'function hell { ' is defined, which contains a 'LOCAL HELLO=world' line and an 'echo \$HELLO' line, followed by a closing brace '}'. The search results are highlighted in green. At the bottom of the terminal, a status bar shows the current file 'lab09.shr', search scope 'All', search mode 'L3', and search engine '(Shell-script[bash] Isearch)'. Below the status bar, the search results are displayed as 'Wrapped I-search: hell'.

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

U:~-- #lab09.shr All L3 (Shell-script[bash] Isearch)
Wrapped I-search: hell

Рис. 24: Поиск по файлу

9.2. Переключайтесь между результатами поиска, нажимая C-s.

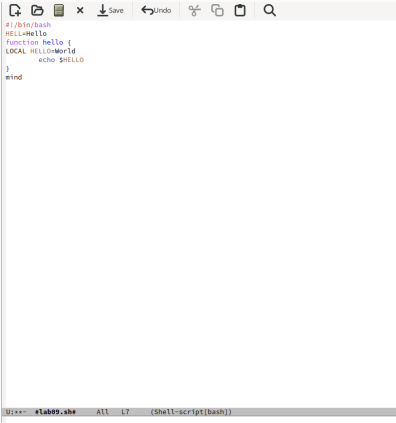


```
#!/bin/bash
HELL=Hello
function hello {
  LOCAL HELLO=world
  echo $HELLO
}
```

The screenshot shows a terminal window with a light gray title bar containing icons for file operations (open, save, delete, copy, paste, search) and a search bar. The terminal content displays a shell script with a function named 'hello' that prints 'world'. The prompt is 'U:*** #lab09.sh#'. The status bar at the bottom shows 'All L6 (Shell-script(bash))' and 'Quit'.

Рис. 25: Переход между результатами поиска

9.3. Выйдите из режима поиска, нажав C-g.




The screenshot shows a terminal window with a light gray title bar and a standard Linux-style toolbar. The terminal content displays a shell script with syntax highlighting: a comment, a variable assignment, a function definition, and a call to the function. At the bottom of the terminal, a status bar indicates the current session is a shell script named 'lab09.sh' running in a bash shell.

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

U:xx- #lab09.sh All L7 (Shell-script(bash))

Рис. 26: Выход из режима поиска

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.



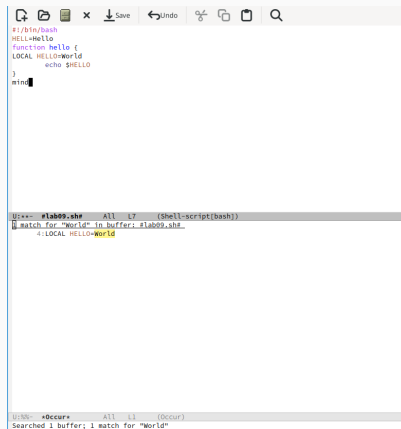
The screenshot shows a Vim editor window with a light gray title bar containing icons for file operations and search. The main editing area has a white background with a blue border on the left. It contains a shell script with the following content:

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

At the bottom of the window, a status bar shows the file name `U:--- #lab09.sh`, the line and column number `All 17`, and the file type `(Shell-script(bash))`. Below the status bar, a search results line is visible: `List lines matching regexp:` followed by a cursor.

Рис. 27: Поиск с заменой (нет)

9.5. Испробуйте другой режим поиска, нажав M-s o. Объясните, чем он отличается от обычного режима?



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

U:~ - #lab09.shr All L7 (Shell-script(bash))

1 match for "World" in buffer: #lab09.shr

4:LOCAL HELLO=world

U:~ - #Occurs All L1 (Occurs)

Searched 1 buffer; 1 match for "World"

Рис. 28: Альтернативный режим поиска

Контрольные вопросы

1. Кратко охарактеризуйте редактор emacs.

Emacs — один из наиболее мощных и широко распространённых редакторов, используемых в мире UNIX. Написан на языке высокого уровня Lisp.

2. Какие особенности данного редактора могут сделать его сложным для освоения новичком?

Большое разнообразие сложных комбинаций клавиш, которые необходимы для редактирования файла и в принципе для работа с Emacs.

3. Своими словами опишите, что такое буфер и окно в терминологии emacs'a.

Буфер - это объект в виде текста. Окно - это прямоугольная область, в которой отображен буфер.

4. Можно ли открыть больше 10 буферов в одном окне?

Да, можно. Технически можно очень много но лучше не стоит.

5. Какие буферы создаются по умолчанию при запуске emacs?

Emacs использует буферы с именами, начинающимися с пробела, для внутренних целей. Отчасти он обращается с буферами с такими именами особым образом – например, по умолчанию в них не записывается информация для отмены изменений.

6. Какие клавиши вы нажмёте, чтобы ввести следующую комбинацию C-с | и C-с C-|?

“Ctrl + c”, а потом “|” и “Ctrl + c” “Ctrl + |”

7. Как поделить текущее окно на две части?

С помощью команды “Ctrl + x 3” (по вертикали) и “Ctrl + x 2” (по горизонтали).

8. В каком файле хранятся настройки редактора emacs?

Настройки emacs хранятся в файле `.emacs`, который хранится в домашней директории пользователя. Кроме этого файла есть ещё папка `.emacs`.

9. Какую функцию выполняет клавиша “<-” и можно ли её переназначить?

Выполняет функцию стереть, думаю можно переназначить.

10. Какой редактор вам показался удобнее в работе vi или emacs? Поясните почему.

Если научиться пользоваться то на практике будет удобнее emacs, но у него (как и у многих) запутанные интерфейсы и ОЧЕНЬ сильная привязка к горячим клавишам и неоднозначным, хоть и коротким командам. А vi открывается в терминале, и для выполнения большинства операций с файлом достаточно клавиш “i” - редактирование, “Esc” - для выхода из редактора и “w” с “q” для сохранения и выхода (что очень просто и логично).

Выводы по проделанной работе

В ходе выполнения лабораторной работы были изучены основы работы с текстовым редактором emacs и его функционалом.