

Лабораторная работа №9: отчет.

Текстовой редактор etacs.

Евдокимов Максим Михайлович. Группа - НФИбд-01-20.

Содержание

Цель работы	4
Задание	5
Указание к работе	6
Основные термины Emacs	6
Выполнение лабораторной работы	8
Контрольные вопросы	36
Выводы	38
Список литературы	39

Список иллюстраций

1	Открытие emacs	8
2	Создание файла	9
3	Ввод текста	10
4	Сохранение файла	11
5	Редактировать текст	12
6	Выделение строк	13
7	Вставка текста в конец	14
8	Выделение области	15
9	Копирование в буфер обмена	16
10	Вставка из буфера обмена	17
11	Вырезание области в буфер обмена	18
12	Отмена действий	19
13	Переход в начало строки	20
14	Переход в конец строки	21
15	Перемещение в начало буфера курсора	22
16	Перемещение в конец буфера курсора	23
17	Список активных буферов	24
18	Перемещение между буферами	25
19	Закрытие сторонних окон	26
20	Переключение между буферами	27
21	Разделение на 2 буфера	28
22	Разделение на 4 буфера	29
23	Одновременное редактирование нескольких файлов	30
24	Поиск по файлу	31
25	Переход между результатами поиска	32
26	Выход из режима поиска	33
27	Поиск с заменой (нет)	34
28	Альтернативный режим поиска	35

Цель работы

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

Задание

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

Указание к работе

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 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.

Выполнение лабораторной работы

1. Открыть emacs.

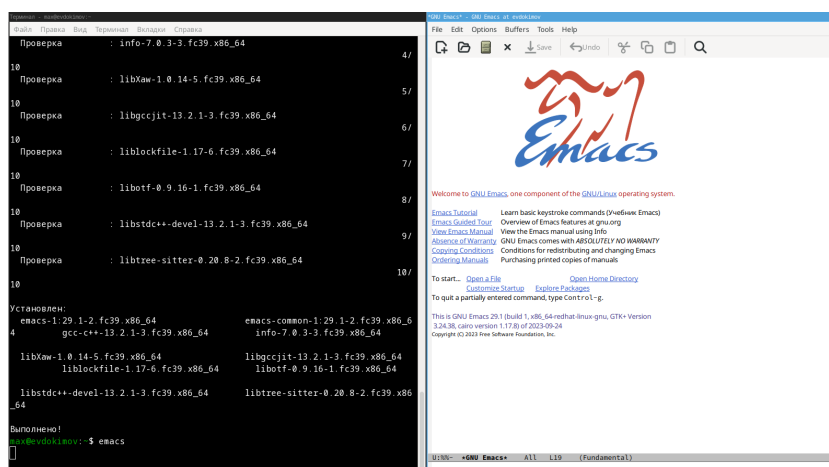


Рис. 1: Открытие emacs

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

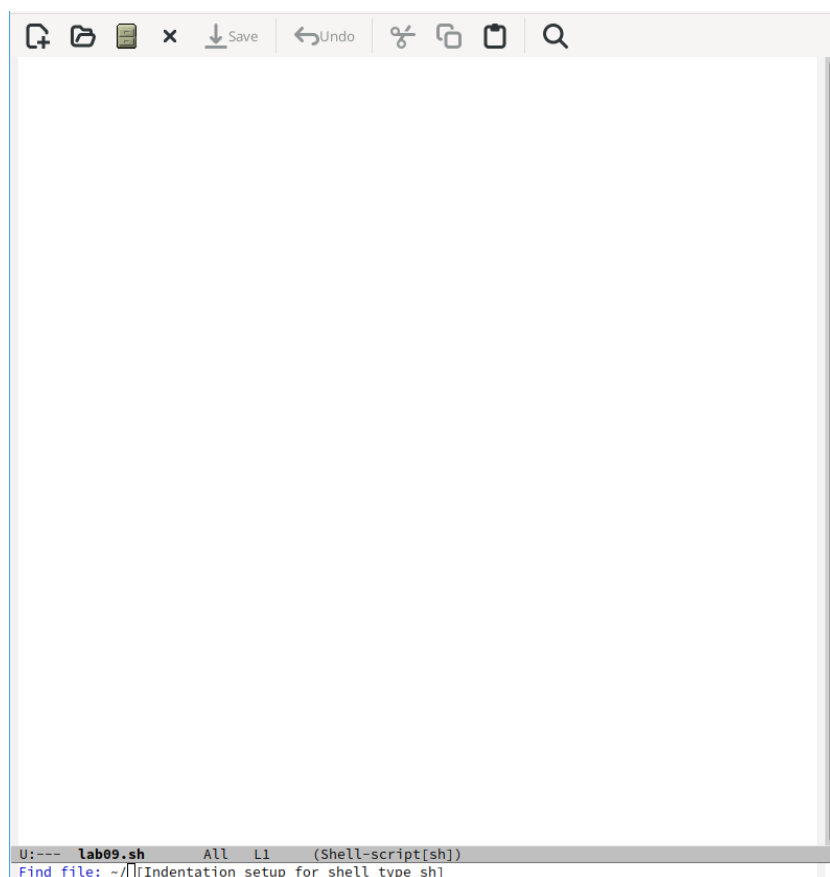
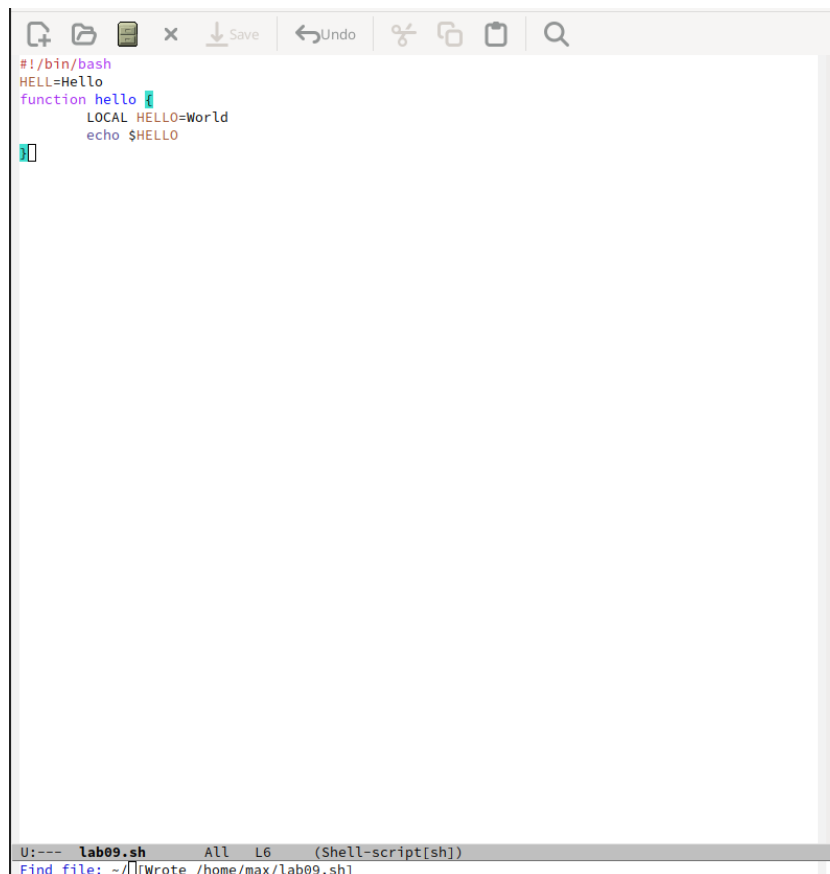


Рис. 2: Создание файла

3. Наберите текст указанный в задание.



The image shows a code editor window with a light gray background. At the top, there 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 written in a monospaced font with syntax highlighting: `#!/bin/bash` (purple), `HELL=Hello` (blue), `function hello` (purple), `{` (blue), `LOCAL HELLO=World` (blue), and `echo $HELLO` (blue). A green cursor is positioned at the end of the `{` line. At the bottom of the editor, a status bar shows the file name `lab09.sh`, the line number `L6`, and the text `(Shell-script[sh])`. Below the status bar, a message reads `Find file: ~/ [Wrote /home/max/lab09.sh]`.

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

Рис. 3: Ввод текста

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

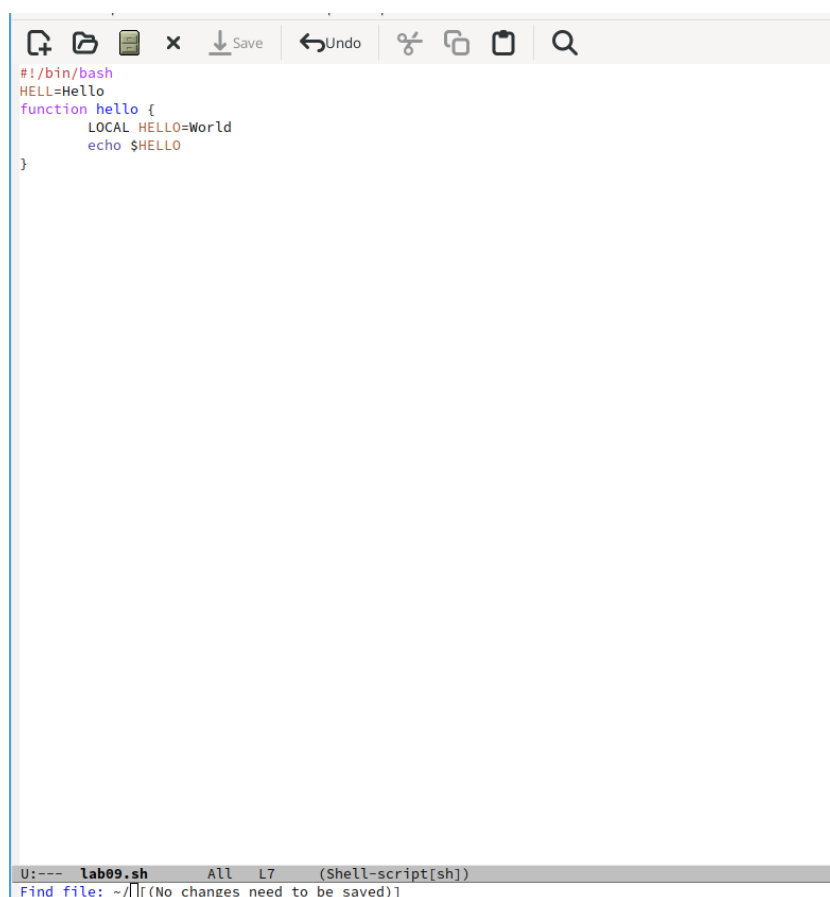
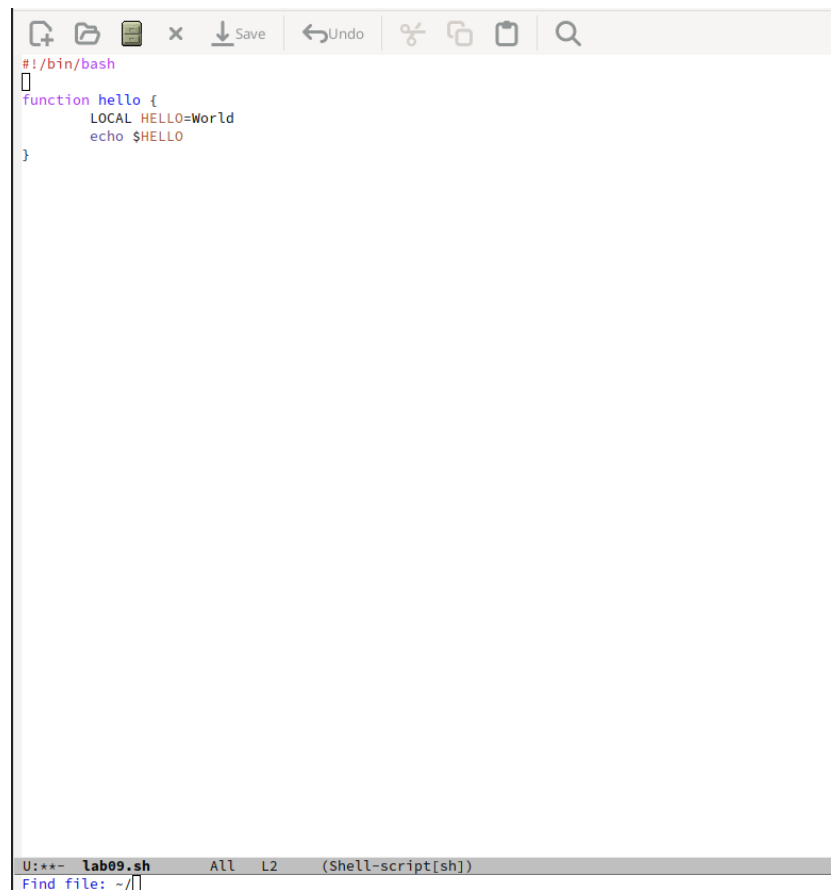


Рис. 4: Сохранение файла

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



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

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

Рис. 5: Редактировать текст

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

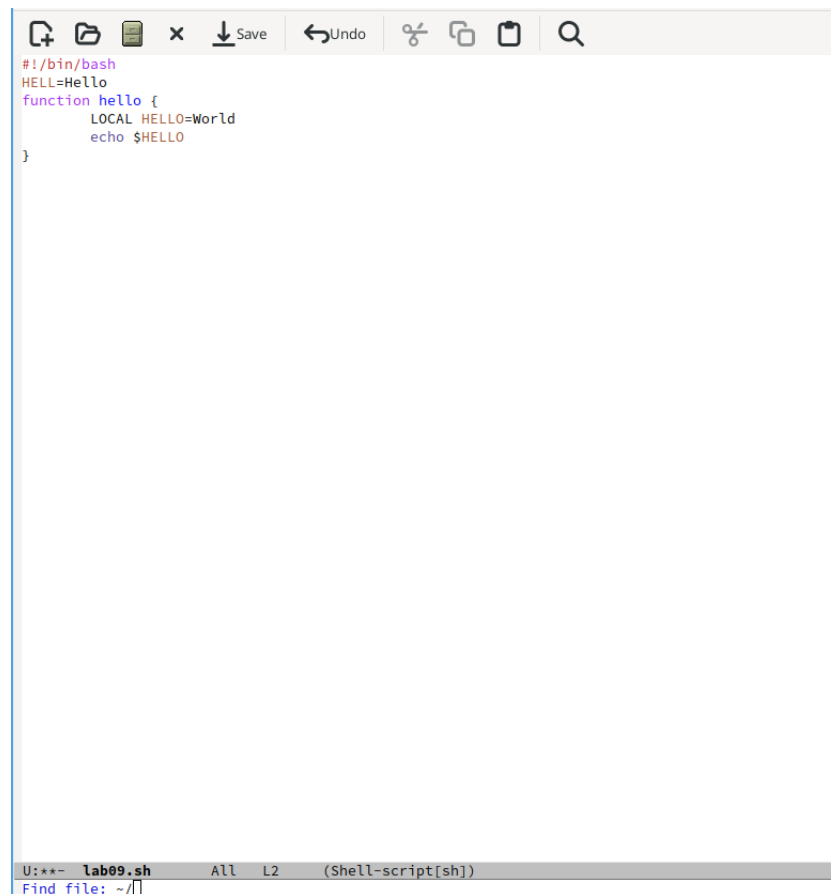


Рис. 6: Веделение строк

5.2. Вставить эту строку в конец файла (C-y).

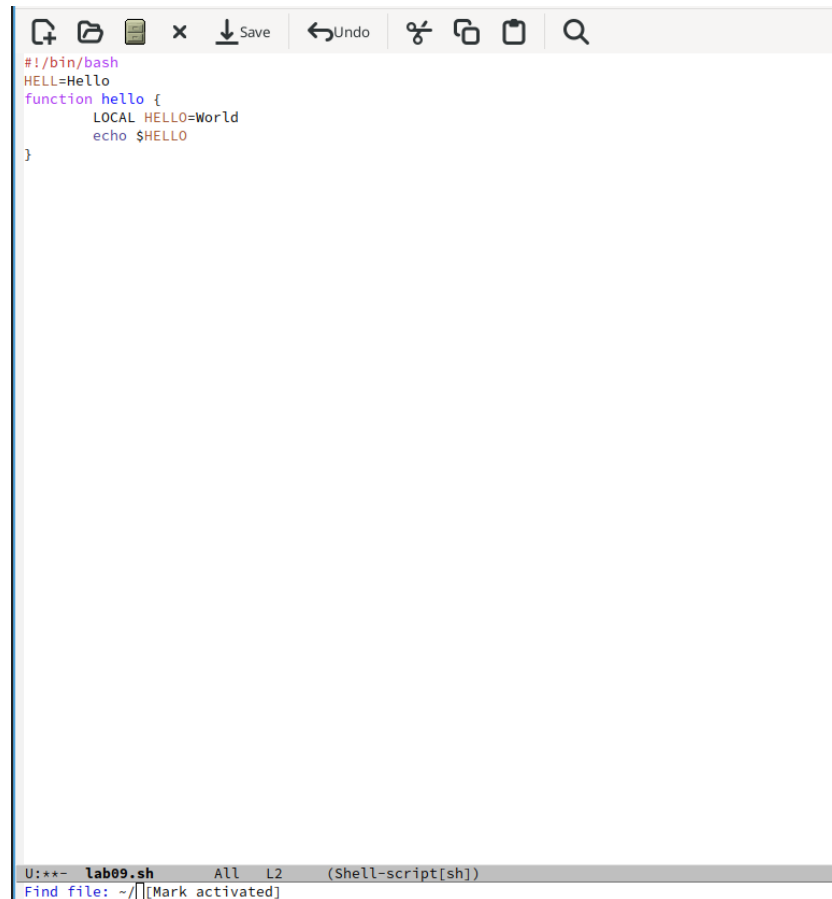
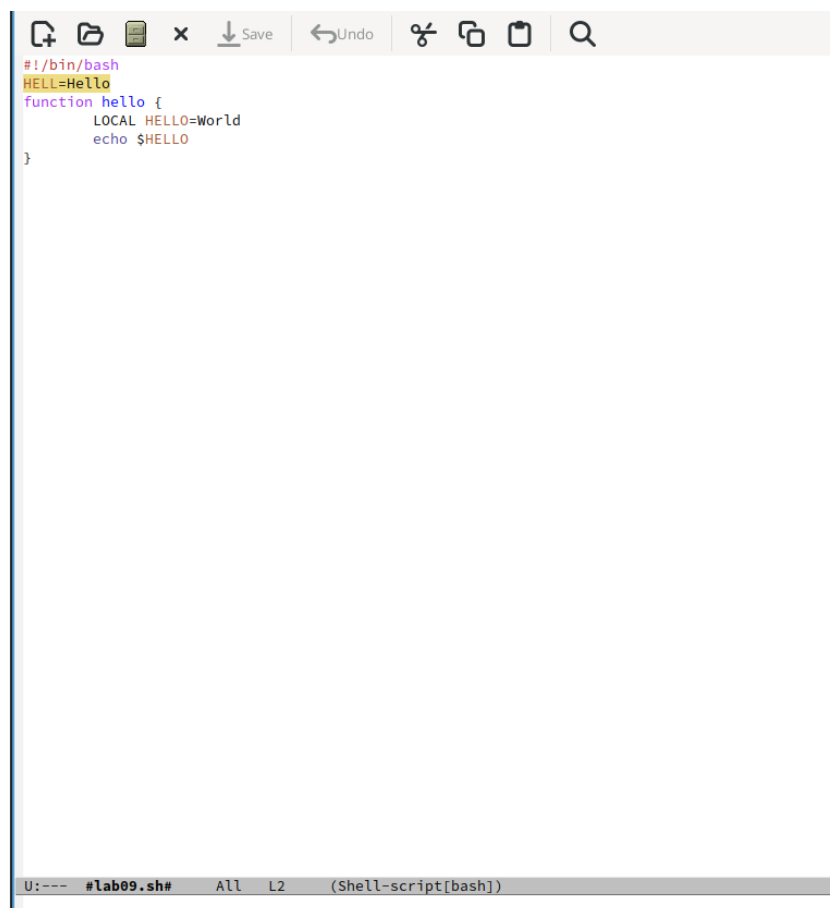


Рис. 7: Вставка текста в конец

5.3. Выделить область текста (C-space).

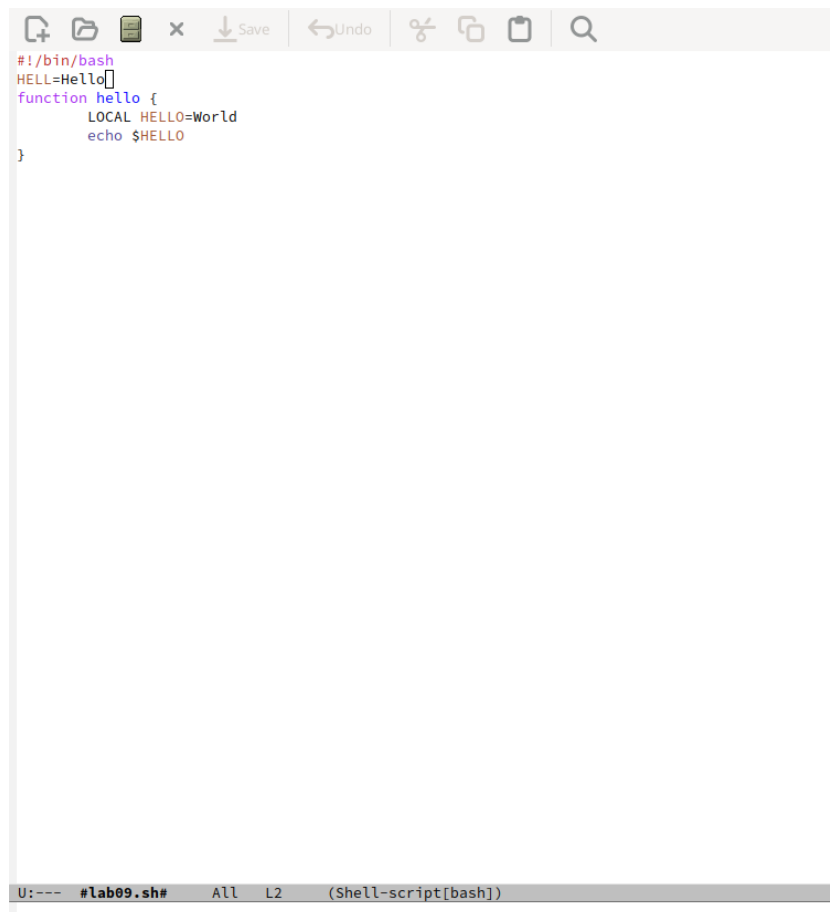


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

U:--- #lab09.sh# All L2 (Shell-script[bash])

Рис. 8: Выделение области

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

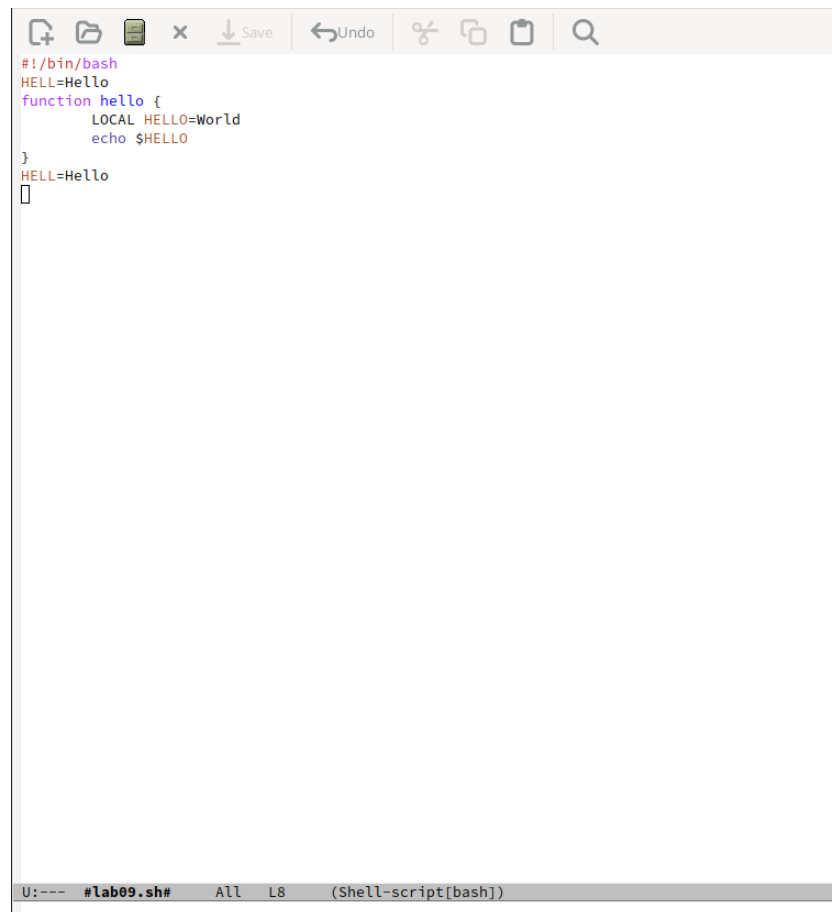


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

U:--- #lab09.sh# All L2 (Shell-script[bash])

Рис. 9: Копирование в буфер обмена

5.5. Вставить область в конец файла.

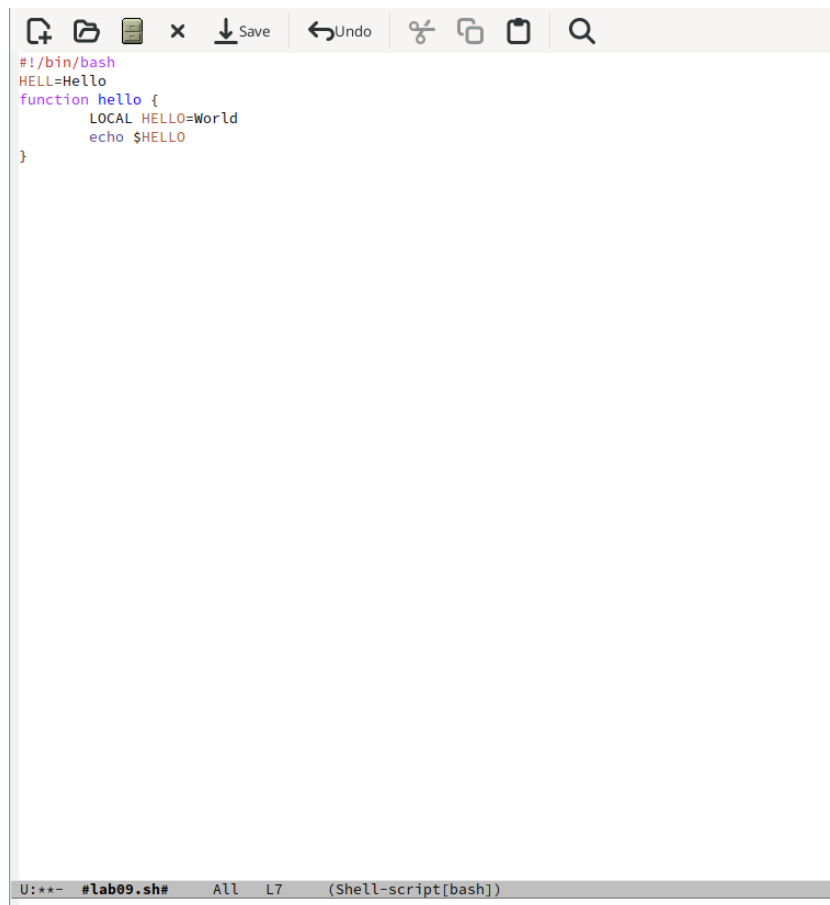


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
HELL=Hello
hello
```

U:--- #lab09.sh# All L8 (Shell-script[bash])

Рис. 10: Вставка из буфера обмена

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



The image shows a code editor window with a light gray background. At the top, there 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
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

At the bottom of the editor, there is a status bar with the text: `U:*** #lab09.sh# All L7 (Shell-script[bash])`.

Рис. 11: Вырезание области в буфер обмена

5.7. Отмените последнее действие (C-/).

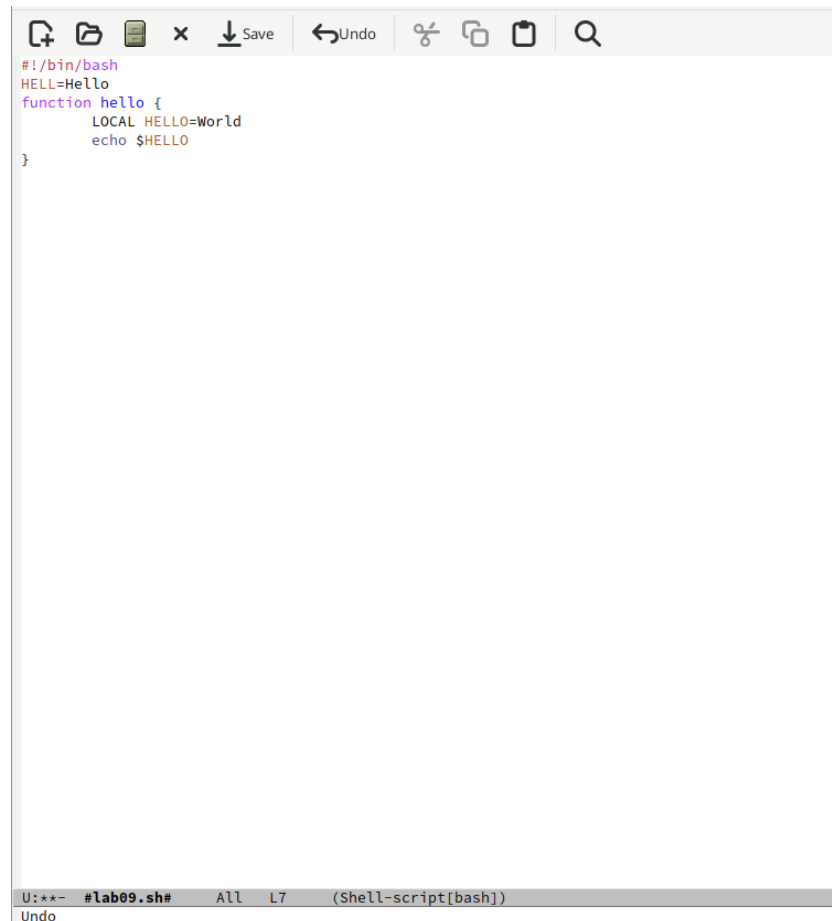
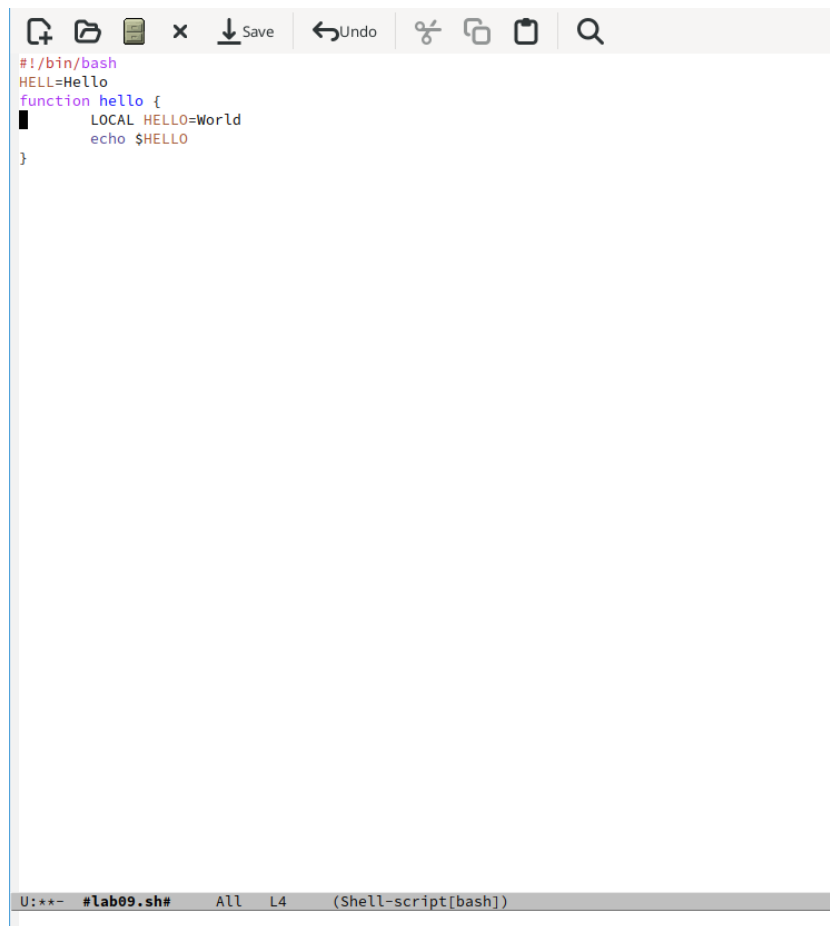


Рис. 12: Отмена действий

6. Научитесь использовать команды по перемещению курсора:

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

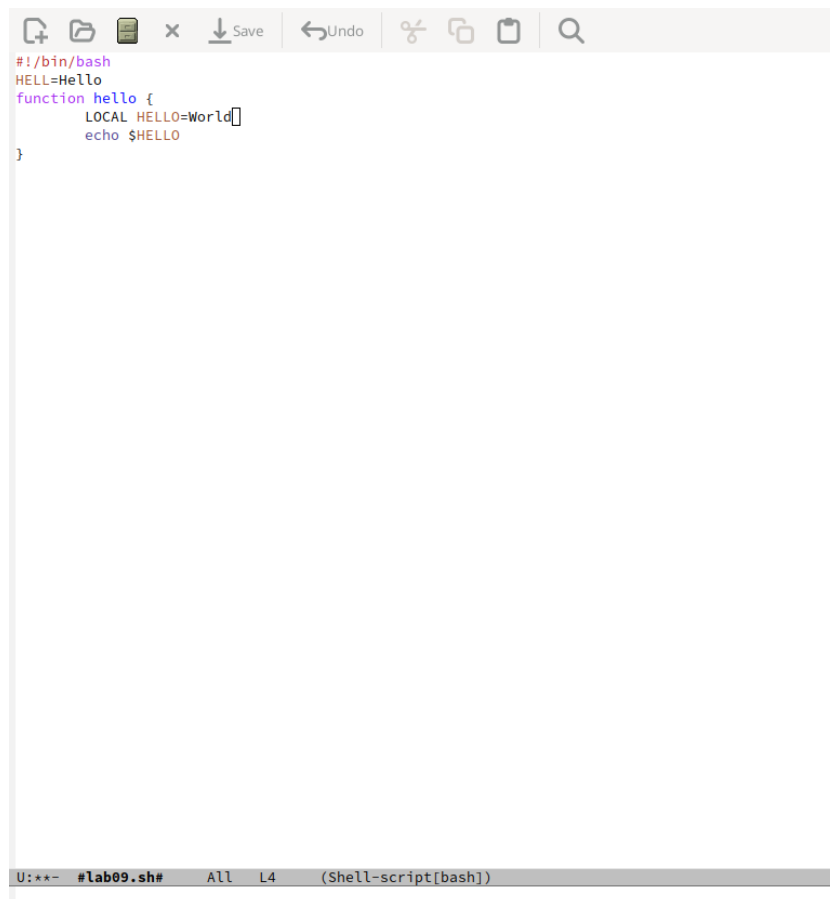


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
```

The screenshot shows a terminal window with a light gray title bar containing icons for file operations (open, save, close, undo, redo, copy, paste, search). The terminal content displays a bash script. The first line is a shebang, the second is an environment variable assignment, and the third is a function definition. The cursor is positioned at the beginning of the third line, before the opening curly brace. The status bar at the bottom indicates the user is 'U', the prompt is '*', the file is '#lab09.sh', and the editor is 'L4' in '(Shell-script[bash])' mode.

Рис. 13: Переход в начало строки

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



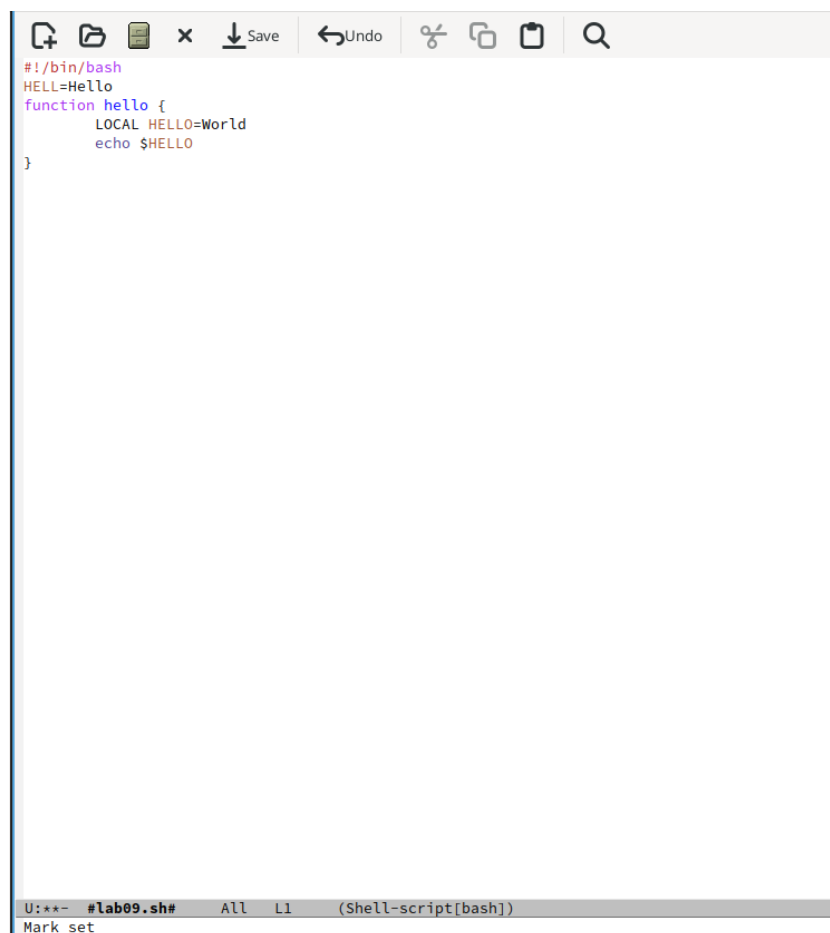
The image shows a code editor window with a light gray background. At the top, there is a toolbar with icons for file operations (new, open, save, close) and editing (undo, redo, copy, paste, search). Below the toolbar, the code is displayed in a monospaced font with syntax highlighting. The code is a bash script snippet:

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

The cursor is positioned at the end of the last line of the function definition. The status bar at the bottom of the editor shows the file path "U:** - #lab09.sh#", the encoding "All", the line number "L4", and the shell type "(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
}
```

U:*** #lab09.sh# All L1 (Shell-script[bash])
Mark set

Рис. 15: Перещение в начало буфера курсора

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

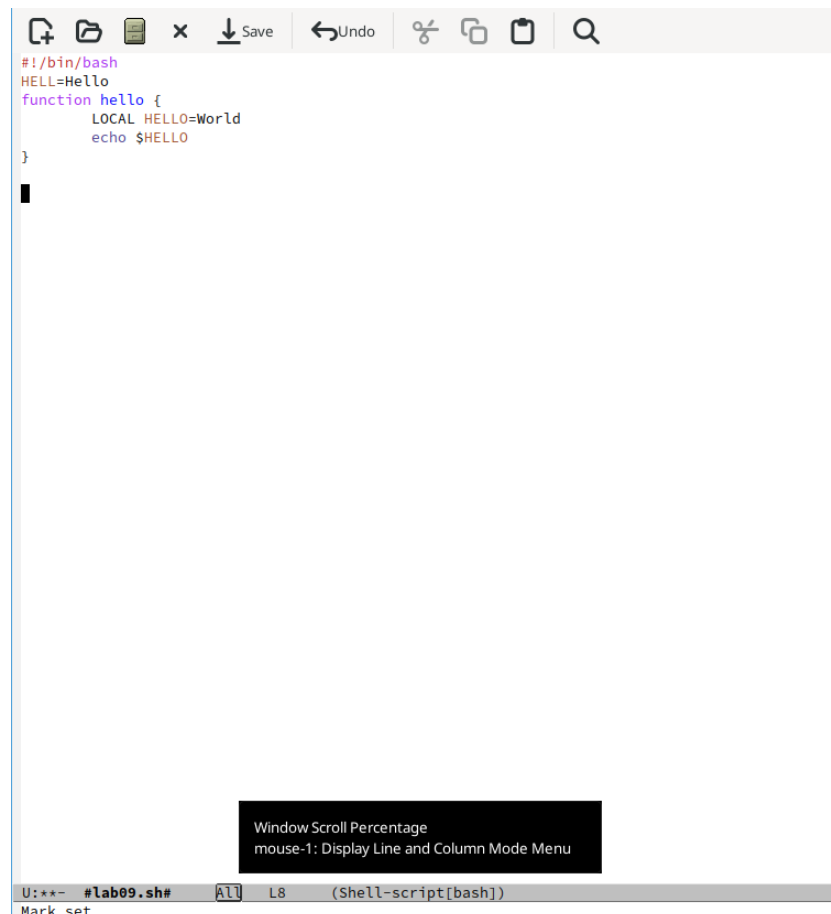


Рис. 16: Перещение в конец буфера курсора

7. Управление буферами:

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

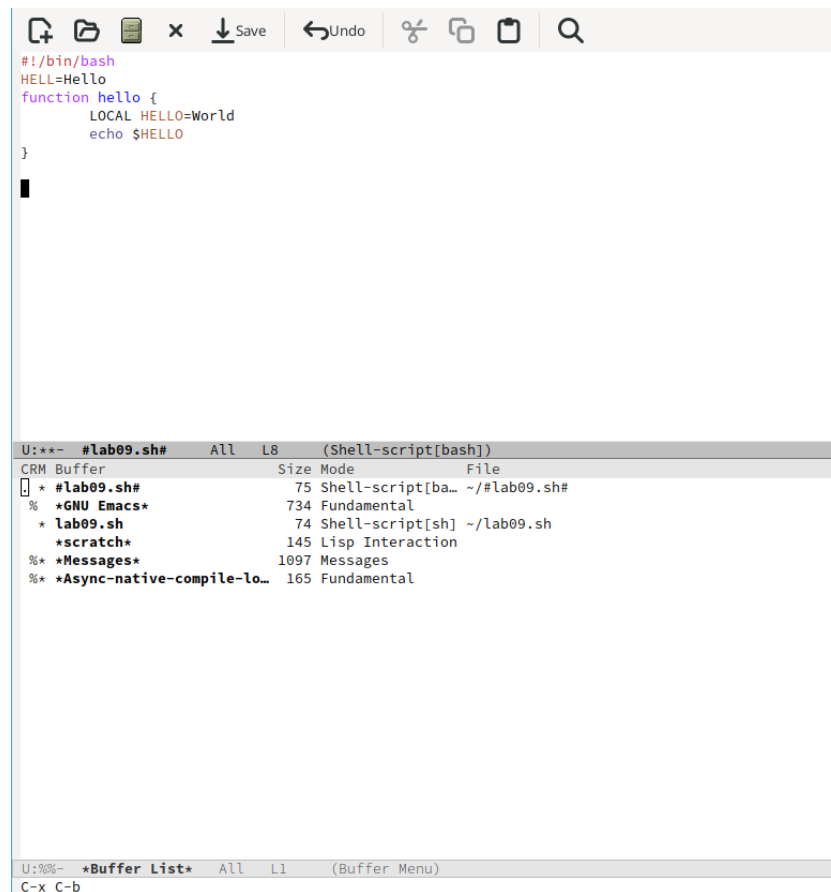


Рис. 17: Список активных буферов

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

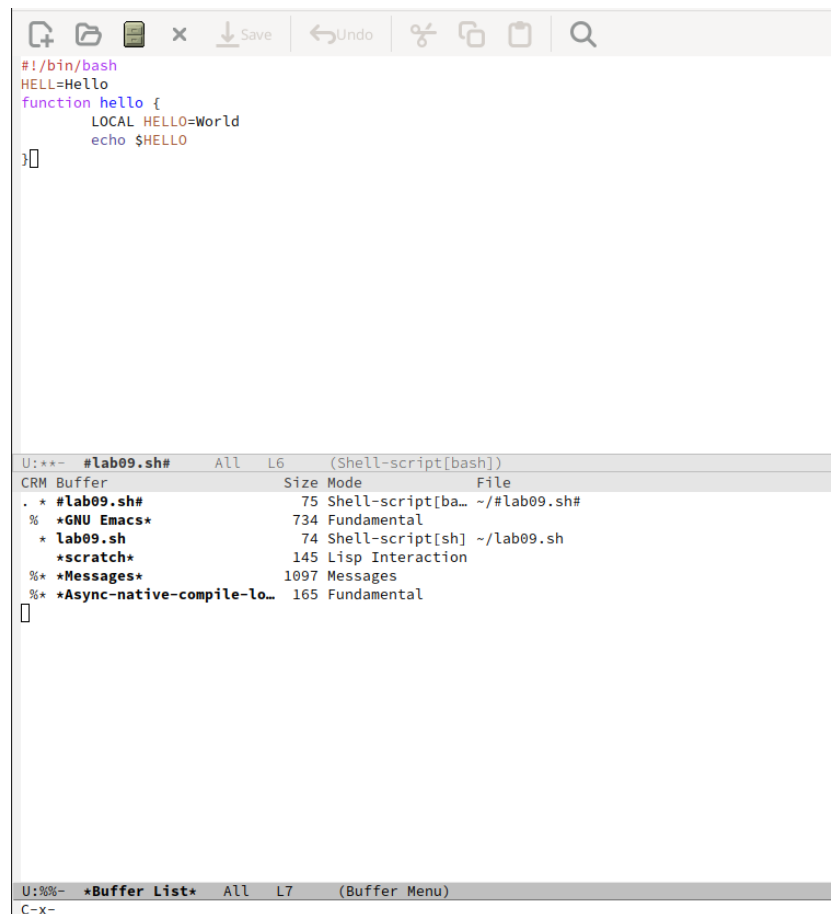


Рис. 18: Перемещение между буферами

7.3. Закройте это окно (C-x 0).



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

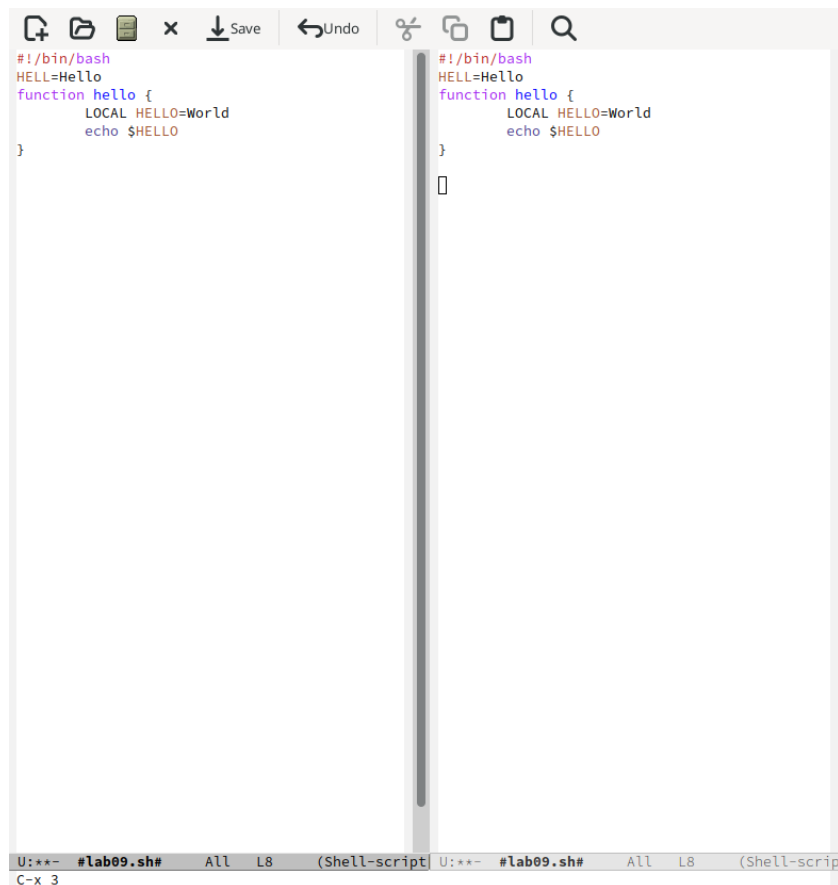


Рис. 20: Переключение между буферами

8. Управление окнами:

8.1. Поделите фрейм на 4 части: разделите фрейм на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по горизонтали (C-x 2) (см. рис. 9.1).

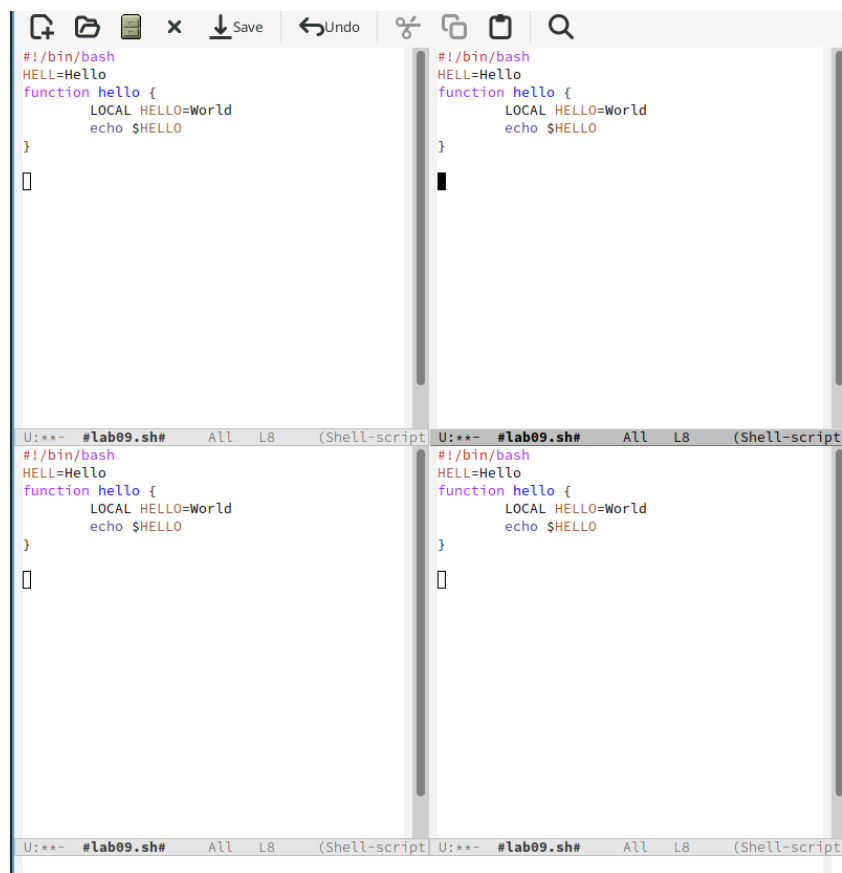


Рис. 21: Разделение на 2 буфера

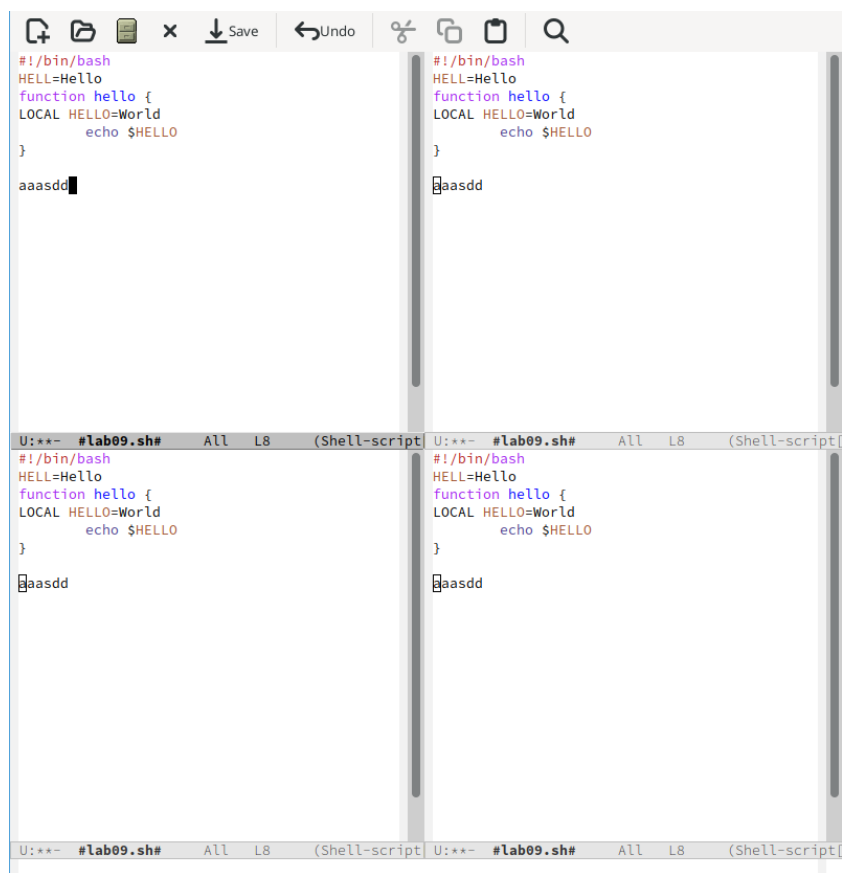


Рис. 22: Разделение на 4 буфера

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

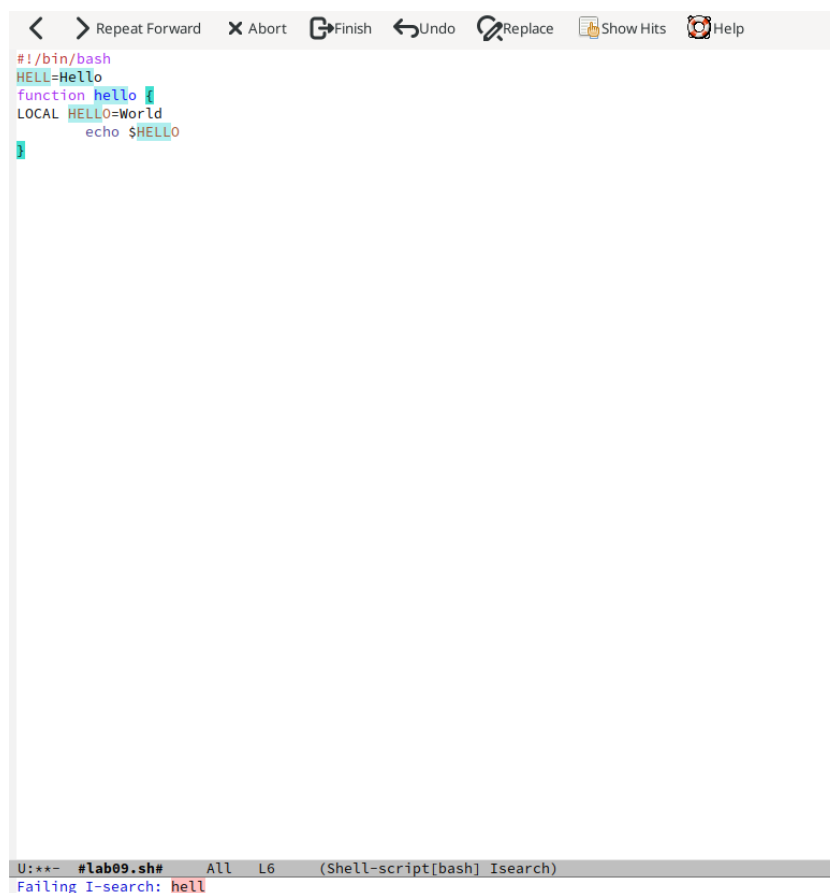


Рис. 23: Одновременное редактирование нескольких файлов

9. Режим поиска:

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

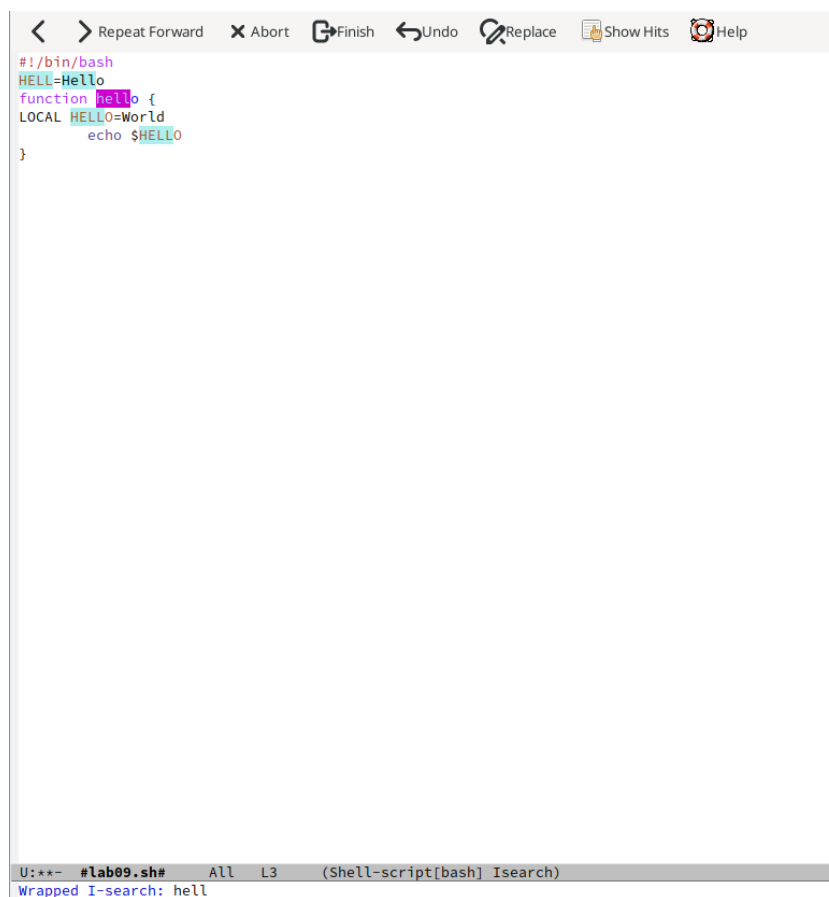
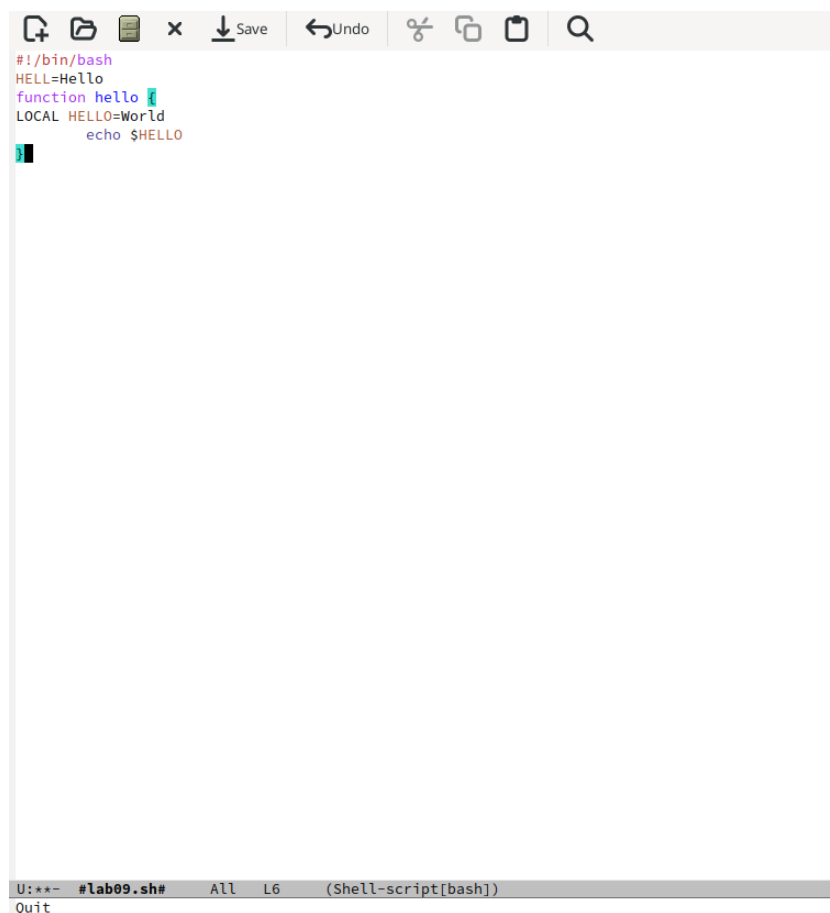


Рис. 24: Поиск по файлу

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



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

U:*** #lab09.sh# All L6 (Shell-script[bash])
Quit
```

Рис. 25: Переход между результатами поиска

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

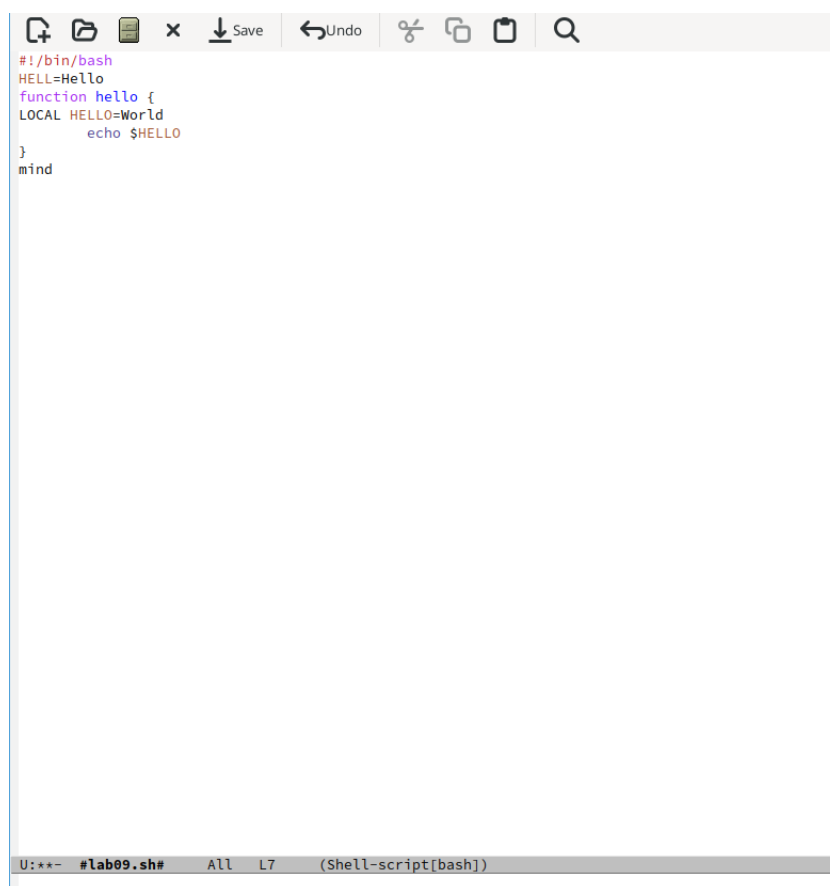


Рис. 26: Выход из режима поиска

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.



Рис. 27: Поиск с заменой (нет)

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

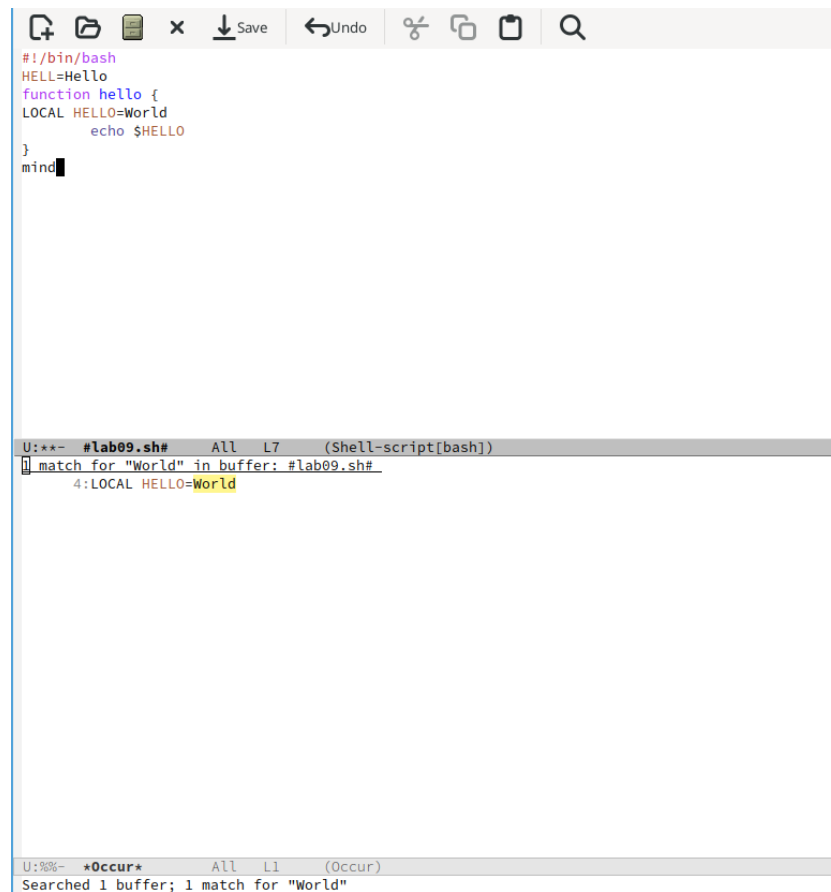


Рис. 28: Альтернативный режим поиска

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'а.

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-c | и C-c 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) запутанный интерфейс 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

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

Список литературы

1. Лабораторная работа №9
2. emacs manual
3. Краткий справочник emacs
4. Статья по работе с emacs