

Отчет по лабораторной работе №9

дисциплина: операционные системы

Шмаков Максим Павлович

Содержание

Цель работы	5
Задание	6
Выполнение лабораторной работы	8
Выводы	38
Контрольные вопросы	39

Список иллюстраций

0.1. рис. 1	8
0.2. рис. 2	9
0.3. рис. 3	9
0.4. рис. 4	10
0.5. рис. 5	10
0.6. рис. 6	11
0.7. рис. 7	12
0.8. рис. 8	13
0.9. рис. 9	14
0.10. рис. 10	15
0.11. рис. 11	16
0.12. рис. 12	17
0.13. рис. 13	18
0.14. рис. 14	19
0.15. рис. 15	20
0.16. рис. 16	21
0.17. рис. 17	22
0.18. рис. 18	23
0.19. рис. 19	24
0.20. рис. 20	25
0.21. рис. 22	27
0.22. рис. 23	28
0.23. рис. 24	29
0.24. рис. 25	30
0.25. рис. 26	31
0.26. рис. 27	32
0.27. рис. 28	33
0.28. рис. 29	33
0.29. рис. 30	34
0.30. рис. 31	35
0.31. рис. 32	36
0.32. рис. 33	36
0.33. рис. 34	37

Список таблиц

Цель работы

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

Задание

Основные команды emacs 1. Открыть emacs. 2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f). 3. Наберите текст: `#!/bin/bash HELLO=Hello`
`function hello { LOCAL HELLO=World echo $HELLO } echo $HELLO hello` 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. 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). 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). 8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста. 9. 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. Объясните, чем он отличается от обычного режима?

Выполнение лабораторной работы

Основные команды emacs 1. Открыть emacs. (рис. [-@fig:001]) (рис. [-@fig:002])

A terminal window with a dark background. The prompt is [mpshmakov@fedora ~]\$ and the command emacs has been entered.

```
[mpshmakov@fedora ~]$ emacs
```

Рис. 0.1.: рис. 1

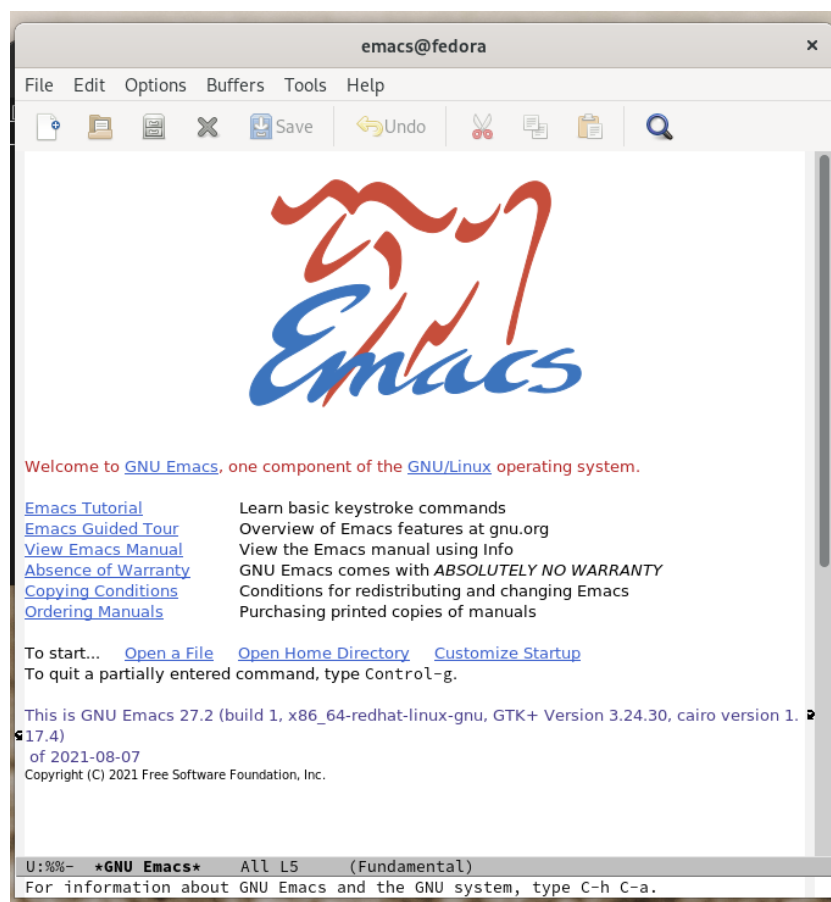


Рис. 0.2.: рис. 2

2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f). (рис. [-@fig:003]) (рис. [-@fig:004])

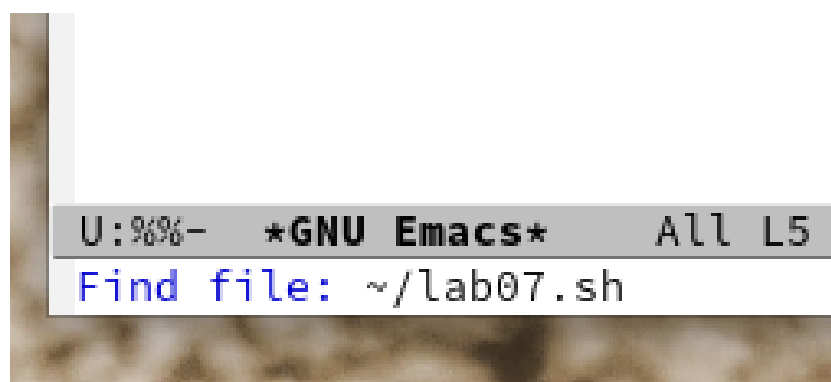


Рис. 0.3.: рис. 3

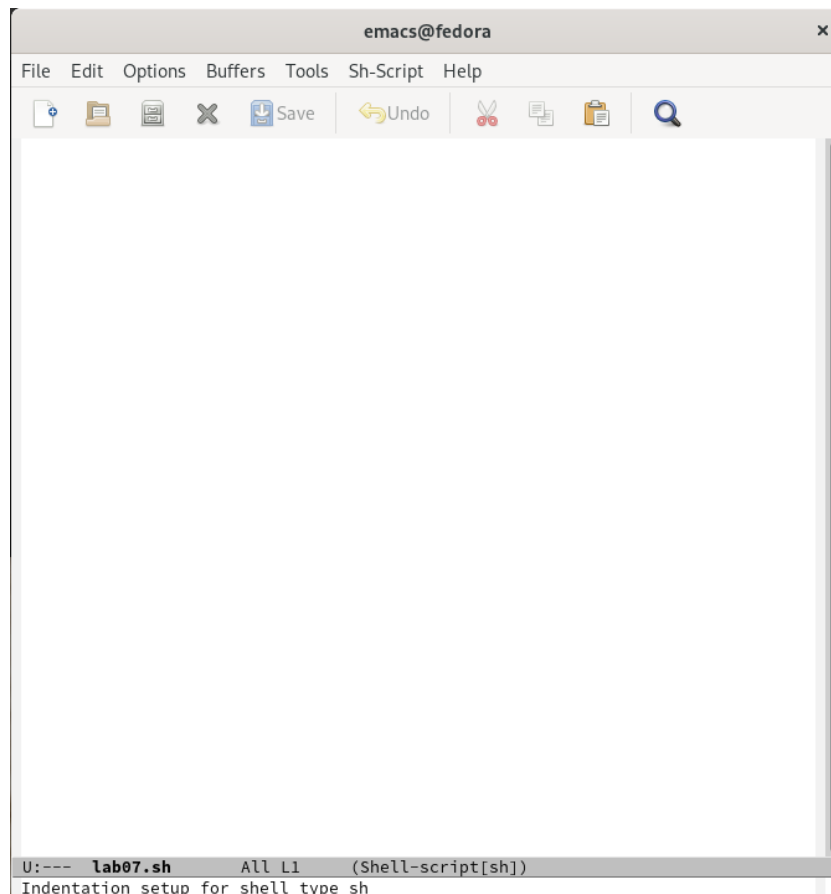


Рис. 0.4.: рис. 4

3. Наберите текст: `#!/bin/bash HELL=Hello function hello { LOCAL HELLO=World echo $HELLO } echo $HELLO hello` (рис. [-@fig:005])

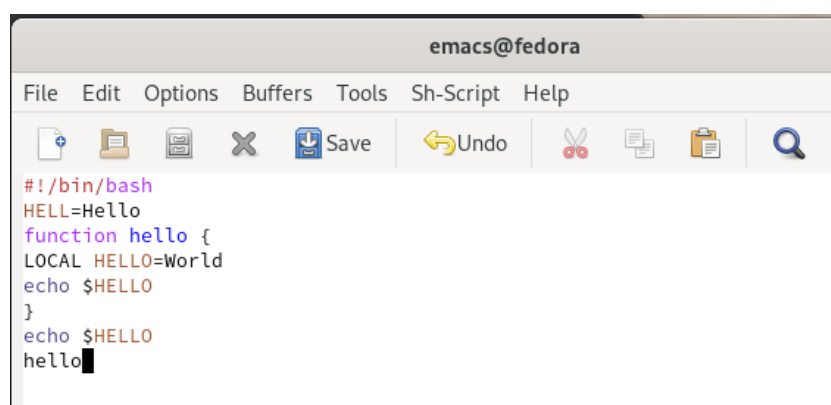


Рис. 0.5.: рис. 5

4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s). (рис. [-@fig:006])

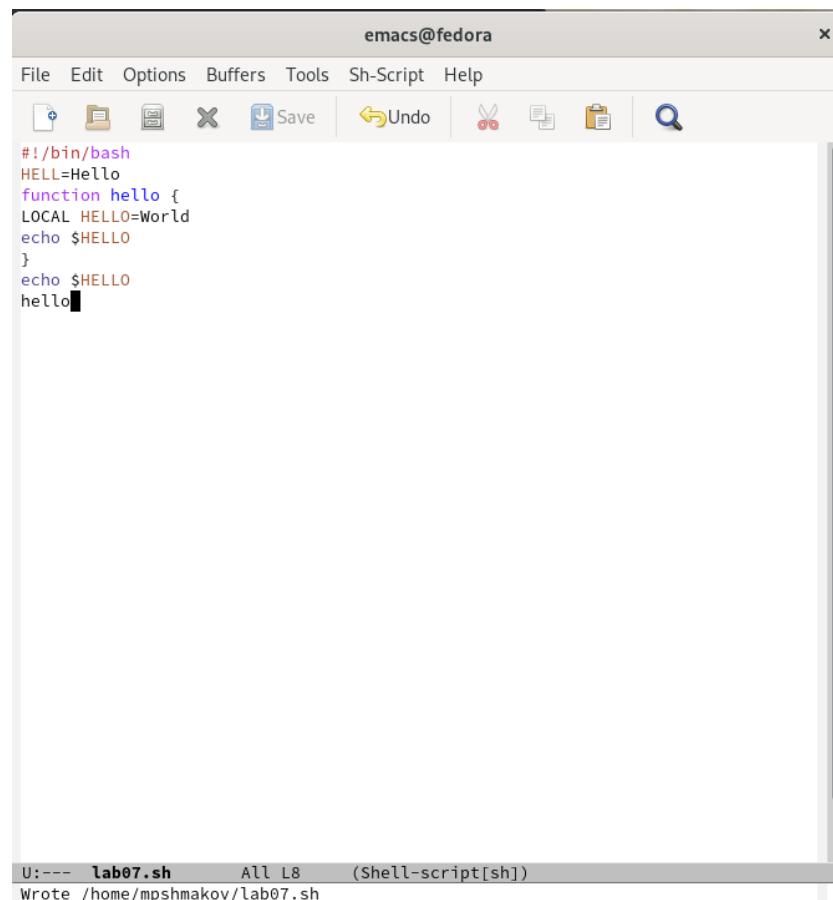


Рис. 0.6.: рис. 6

5. Прodelать с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш. 5.1. Вырезать одной командой целую строку (C-k).

Ставлю курсор в начала строки и использую команду (C-k). (рис. [-@fig:007])

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

Рис. 0.7.: рис. 7

5.2. Вставить эту строку в конец файла (С-у). (рис. [-@fig:008])

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

hello
echo $HELLO
```

Рис. 0.8.: рис. 8

5.3. Выделить область текста (C-space). (рис. [-@fig:009])

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

hello
echo $HELLO
```

Рис. 0.9.: рис. 9

5.4. Скопировать область в буфер обмена (M-w). (рис. [-@fig:010])

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

hello
echo $HELLO
```

Рис. 0.10.: рис. 10

5.5. Вставить область в конец файла. (рис. [-@fig:011])

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

hello
echo $HELLO
$HELLO
```

Рис. 0.11.: рис. 11

5.6. Вновь выделить эту область и на этот раз вырезать её (C-w). (рис. [-@fig:012]) (рис. [-@fig:013])


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

hello
echo $HELLO
$HELLO
```

Рис. 0.12.: рис. 12

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

hello
echo $HELLO
```

Рис. 0.13.: рис. 13

5.7. Отмените последнее действие (C-/). (рис. [-@fig:014])

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

hello
echo $HELLO
$HELLO
```

Рис. 0.14.: рис. 14

6. Научитесь использовать команды по перемещению курсора. 6.1. Переместите курсор в начало строки (C-a). (рис. [-@fig:015])

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

hello
echo $HELLO
$HELLO
```

Рис. 0.15.: рис. 15

6.2. Переместите курсор в конец строки (C-e). (рис. [-@fig:016])

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

hello
echo $HELLO
$HELLO
```

Рис. 0.16.: рис. 16

6.3. Переместите курсор в начало буфера (M-<). (рис. [-@fig:017])

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

hello
echo $HELLO
$HELLO
```

Рис. 0.17.: рис. 17

6.4. Переместите курсор в конец буфера (M->). (рис. [-@fig:018])

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

hello
echo $HELLO
$HELLO
```

Рис. 0.18.: рис. 18

7. Управление буферами. 7.1. Вывести список активных буферов на экран (С-х С-b). (рис. [-@fig:0019])

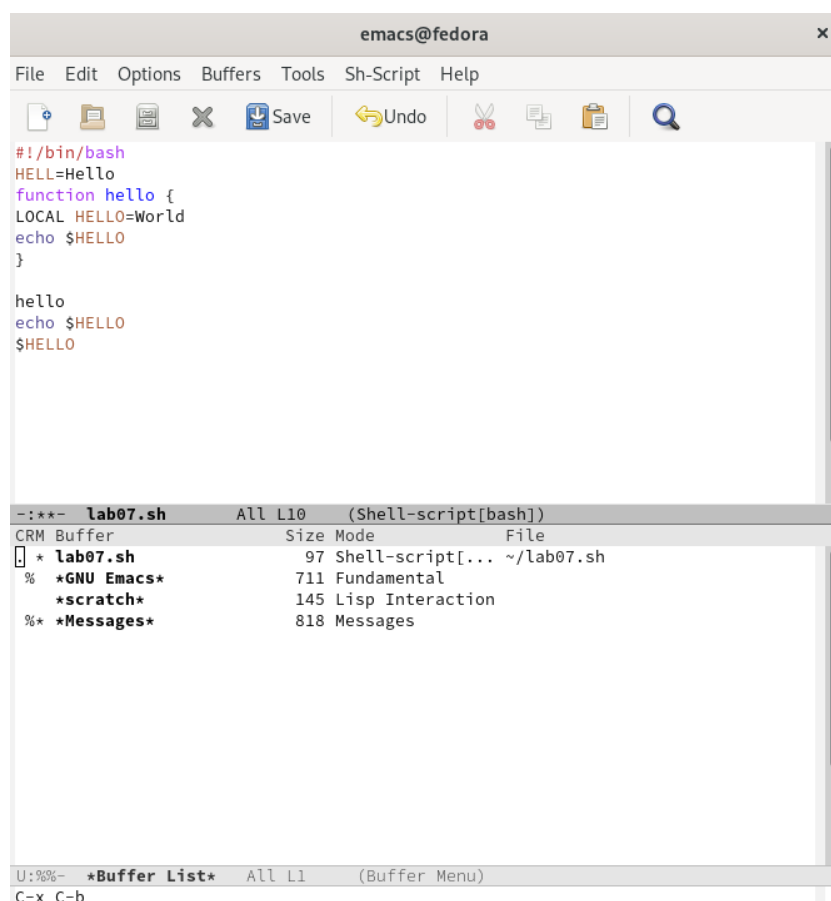


Рис. 0.19.: рис. 19

7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых буферов и переключитесь на другой буфер. (рис. [-@fig:020])

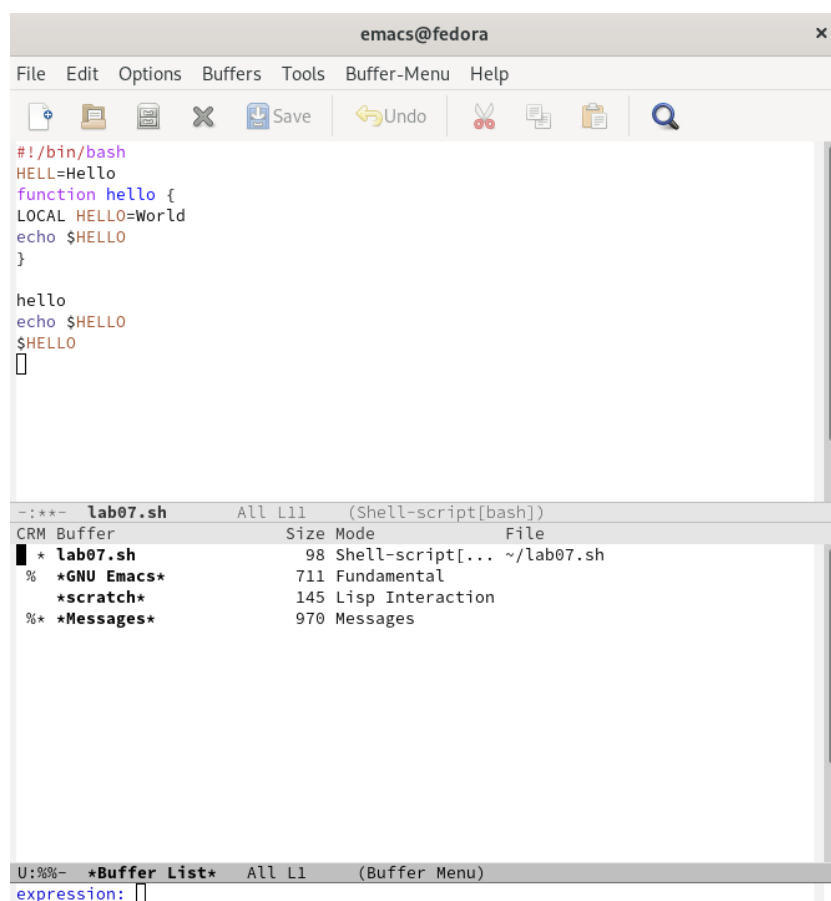
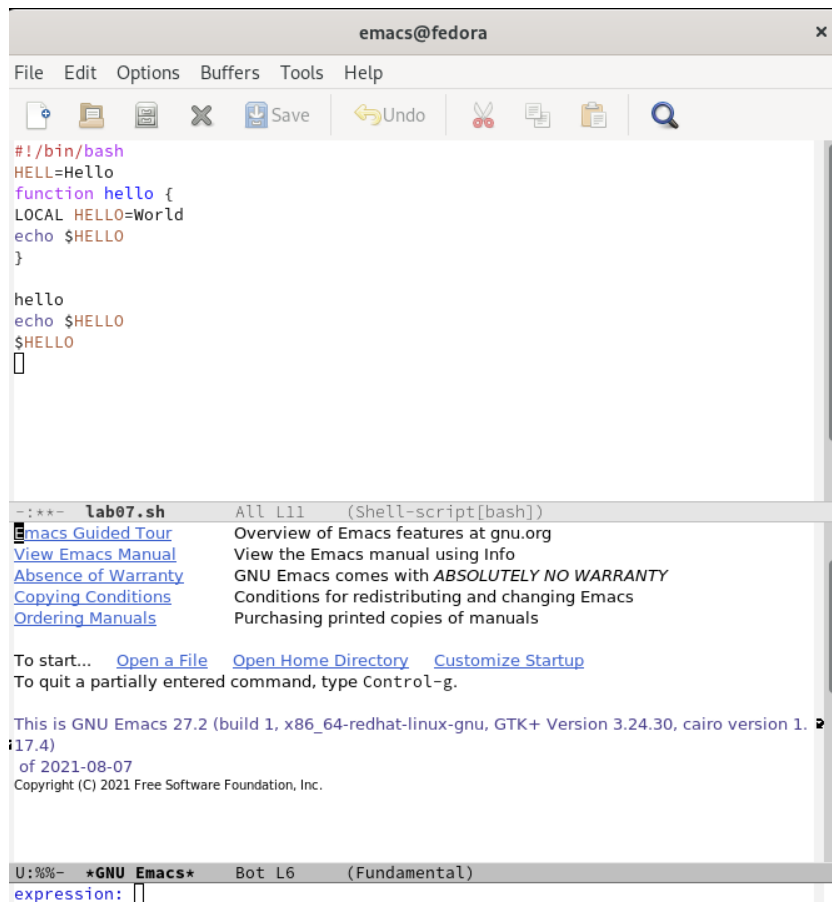


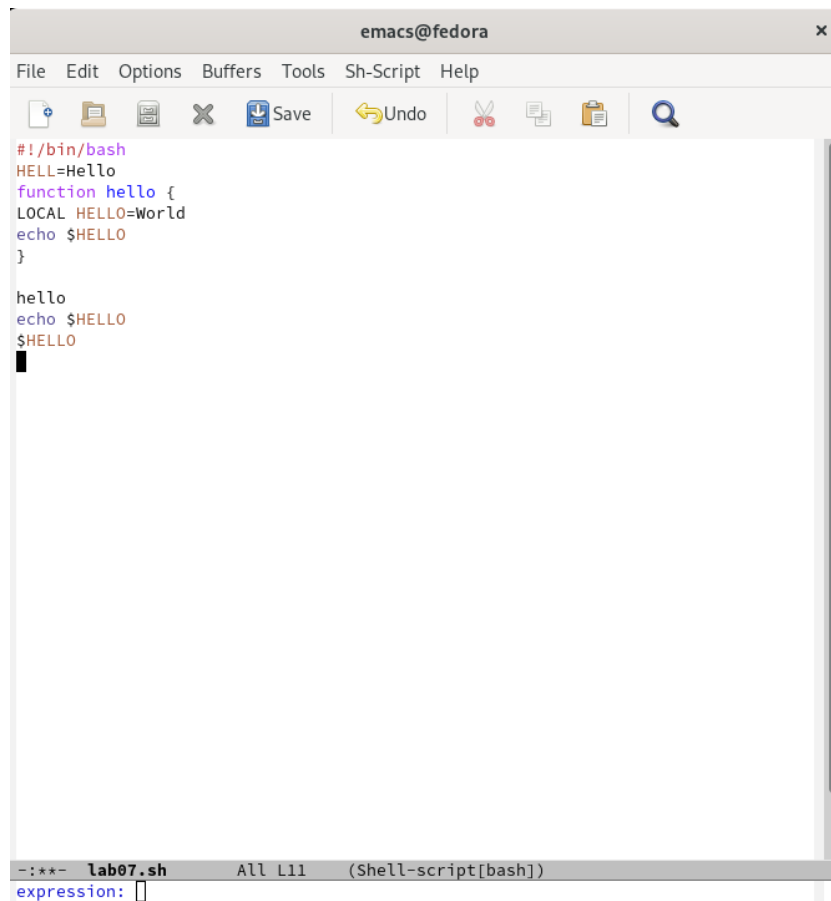
Рис. 0.20.: рис. 20

Навожу курсор на каталог и с помощью enter переключаюсь на него. (рис. [-



@fig:021))

7.3. Закройте это окно (C-x 0). (рис. [-@fig:022])



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

hello
echo $HELLO
$HELLO
```

emacs@fedora

File Edit Options Buffers Tools Sh-Script Help

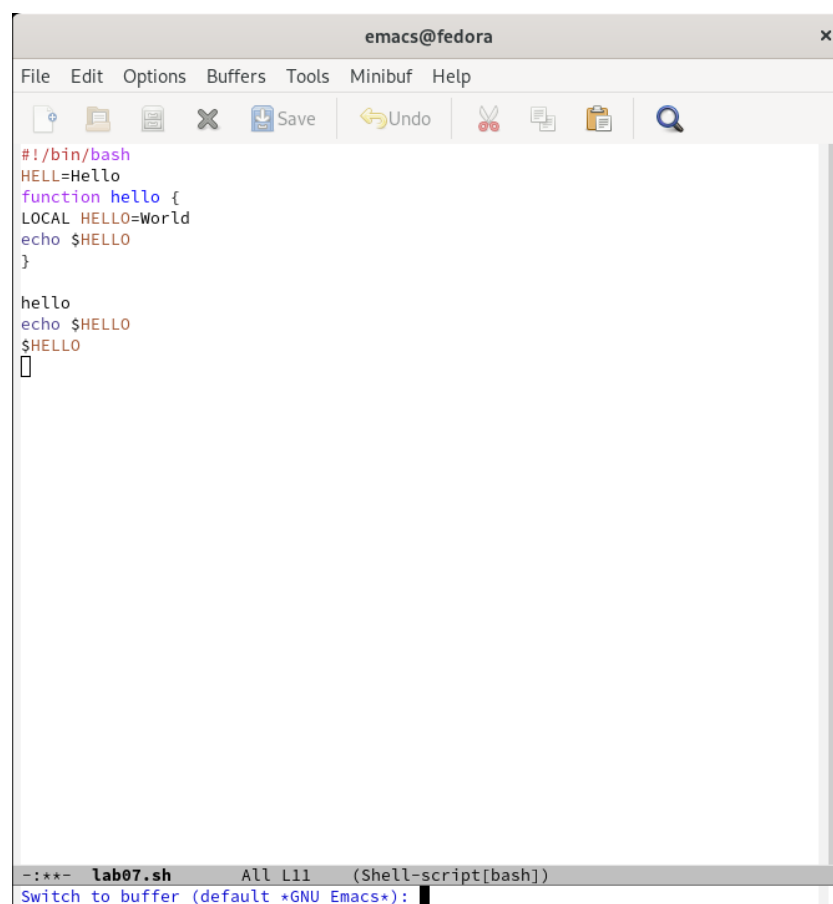
Save Undo

lab07.sh All L11 (Shell-script[bash])

expression:

Рис. 0.21.: рис. 22

7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b). (рис. [-@fig:023]) (рис. [-@fig:024])



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

hello
echo $HELLO
$HELLO
```

Switch to buffer (default *GNU Emacs*):

Рис. 0.22.: рис. 23

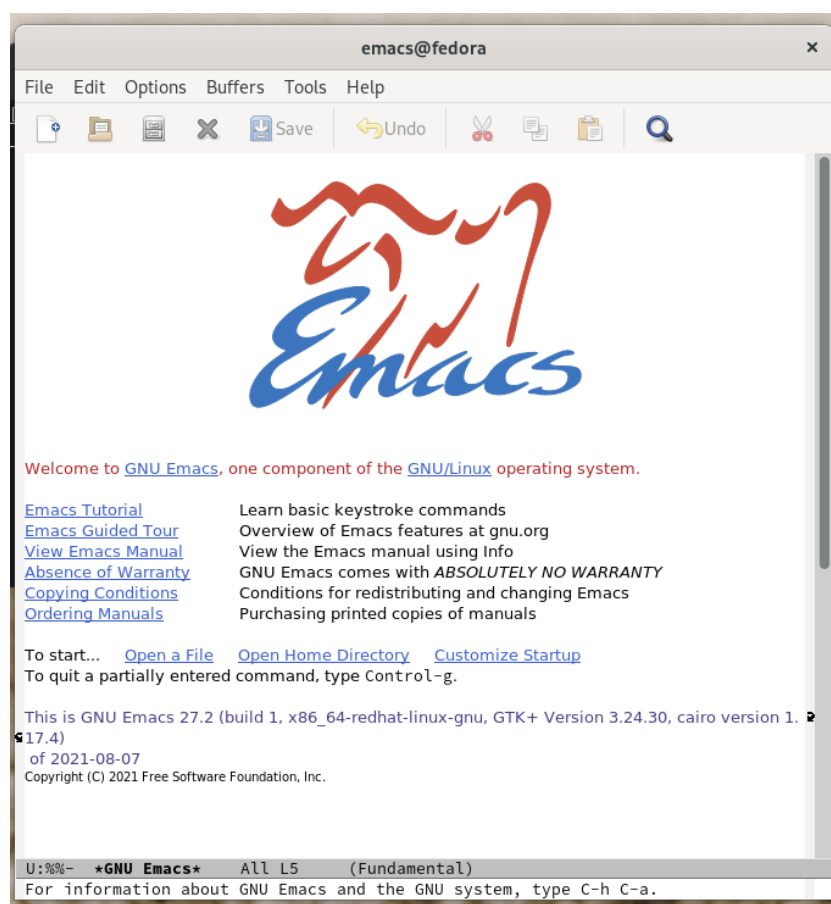


Рис. 0.23.: рис. 24

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) (рис. [-@fig:025]) (рис. [-@fig:026])

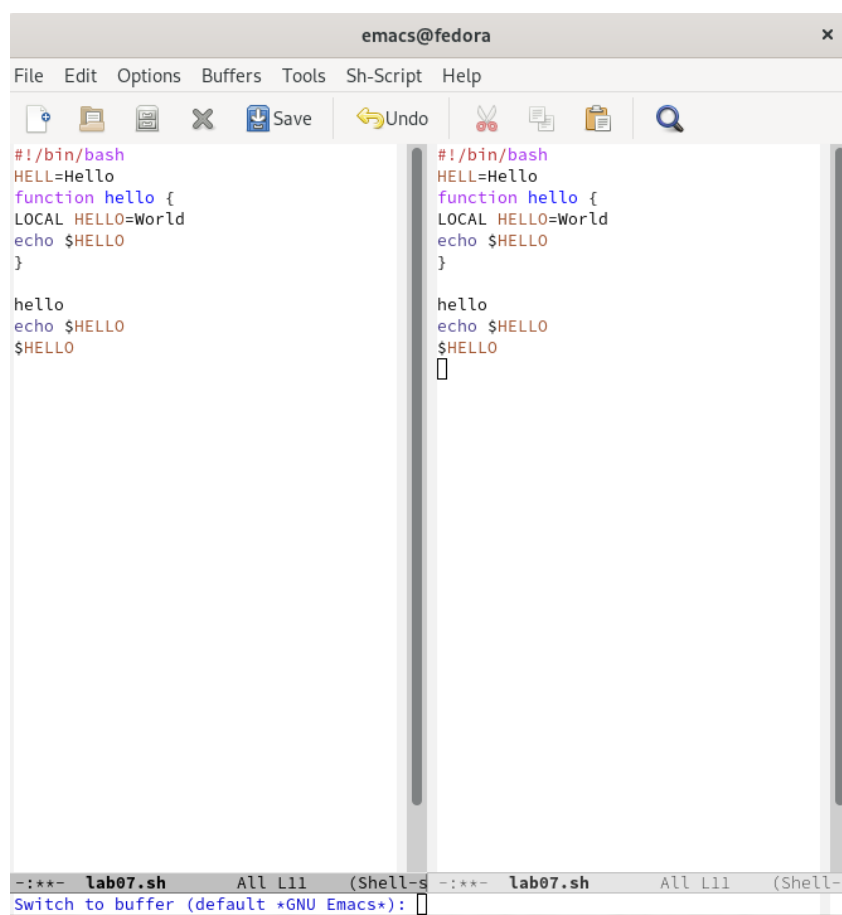


Рис. 0.24.: рис. 25

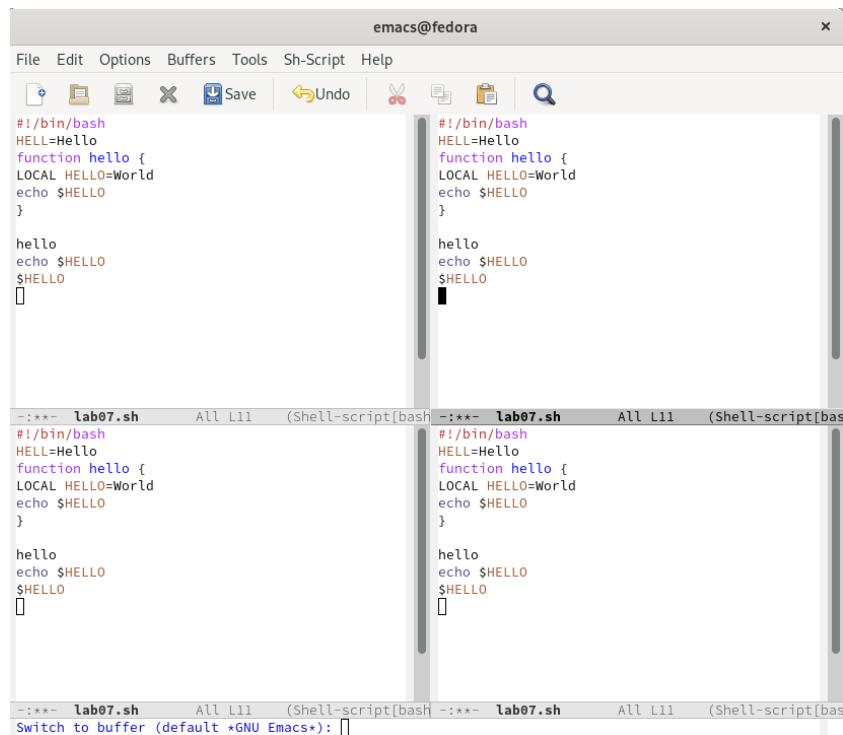


Рис. 0.25.: рис. 26

8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста. (рис. [-@fig:027])

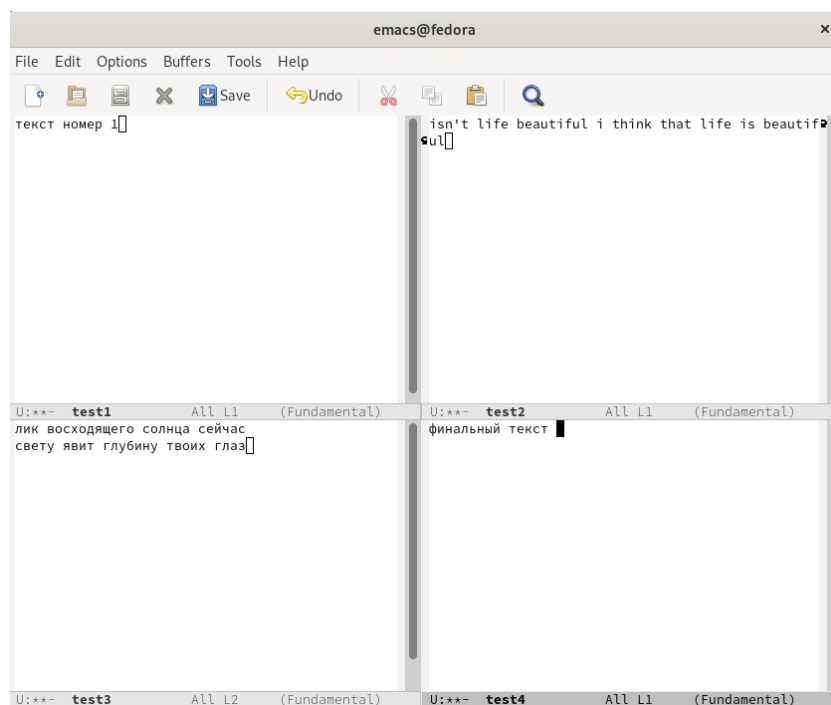


Рис. 0.26.: рис. 27

9. Режим поиска 9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте. (рис. [-@fig:028]) (рис. [-@fig:029])

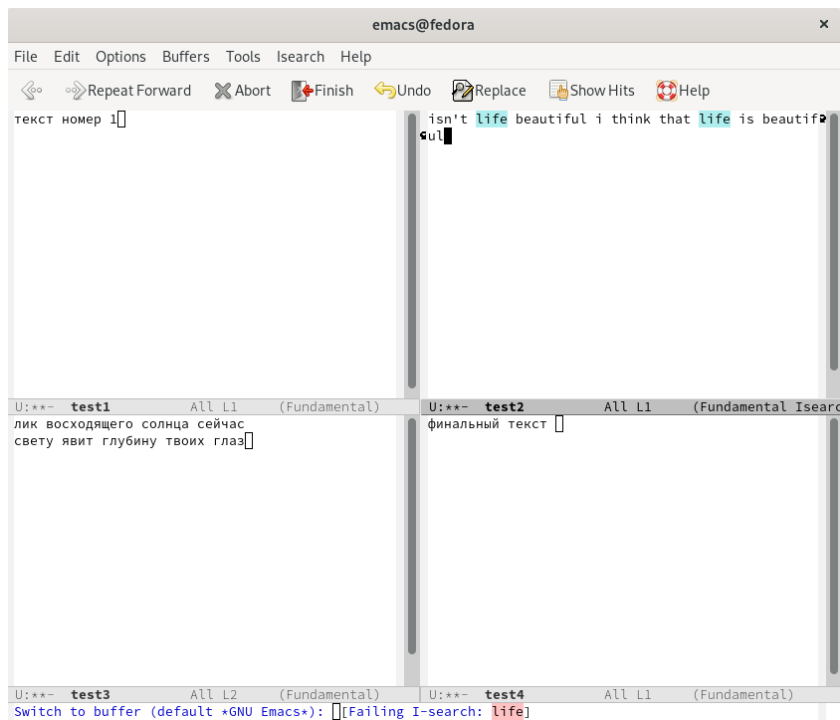


Рис. 0.27.: рис. 28

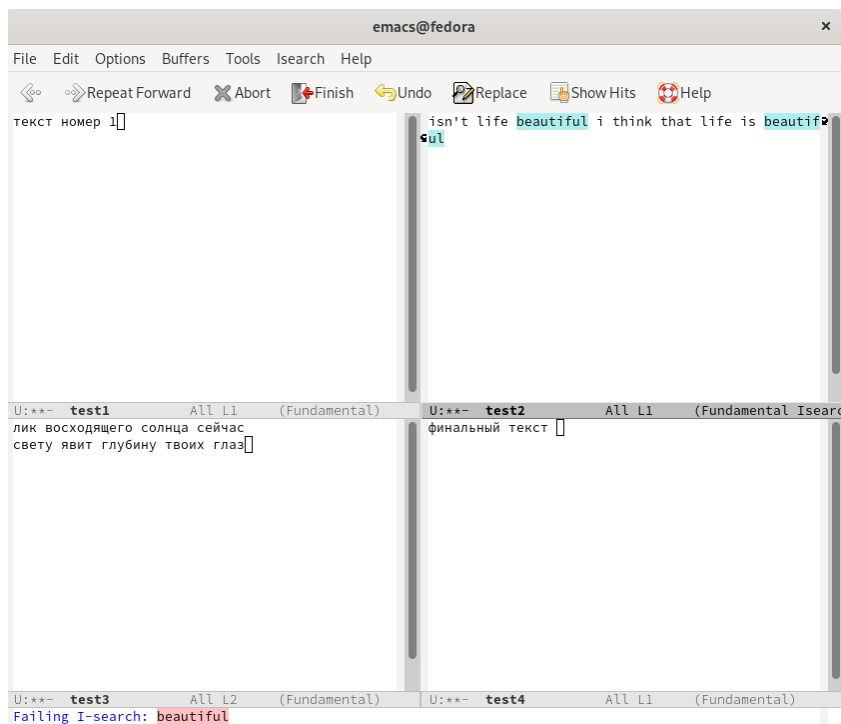


Рис. 0.28.: рис. 29

9.2. Переключайтесь между результатами поиска, нажимая C-s. (рис. [-@fig:030])

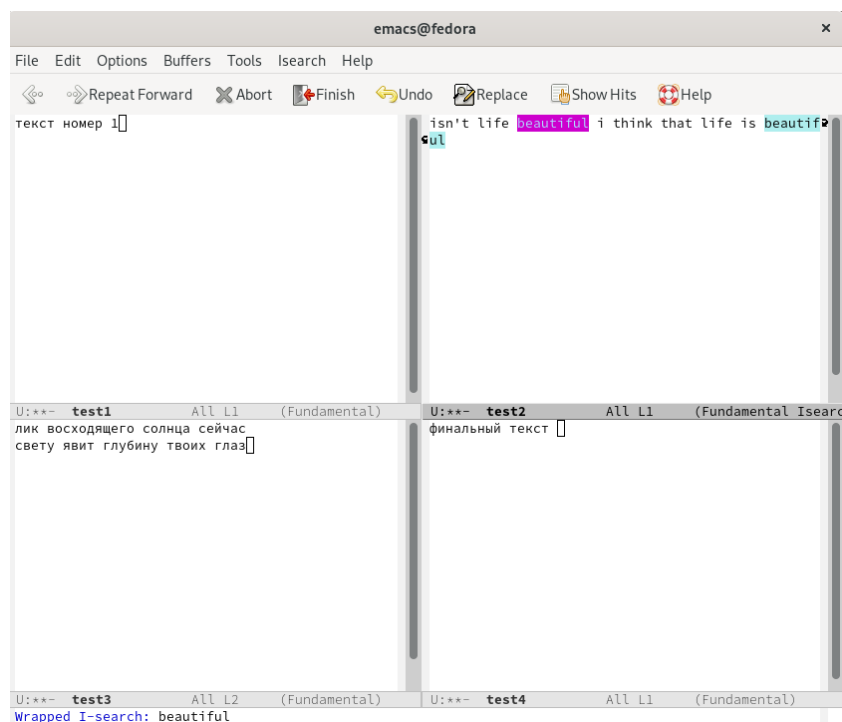


Рис. 0.29.: рис. 30

9.3. Выйдите из режима поиска, нажав C-g. (рис. [-@fig:031])

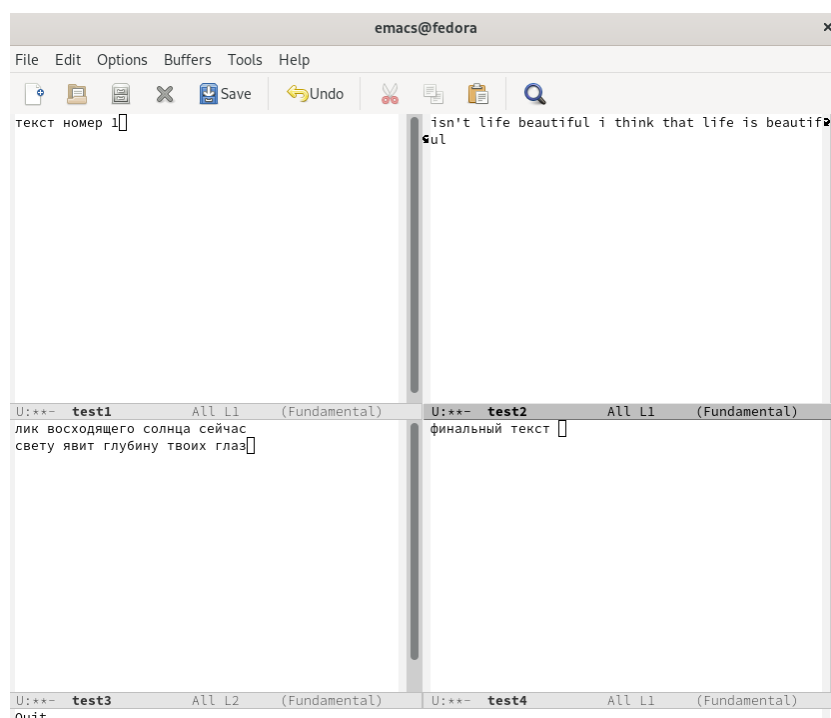


Рис. 0.30.: рис. 31

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. [-@fig:032]) (рис. [-@fig:033])

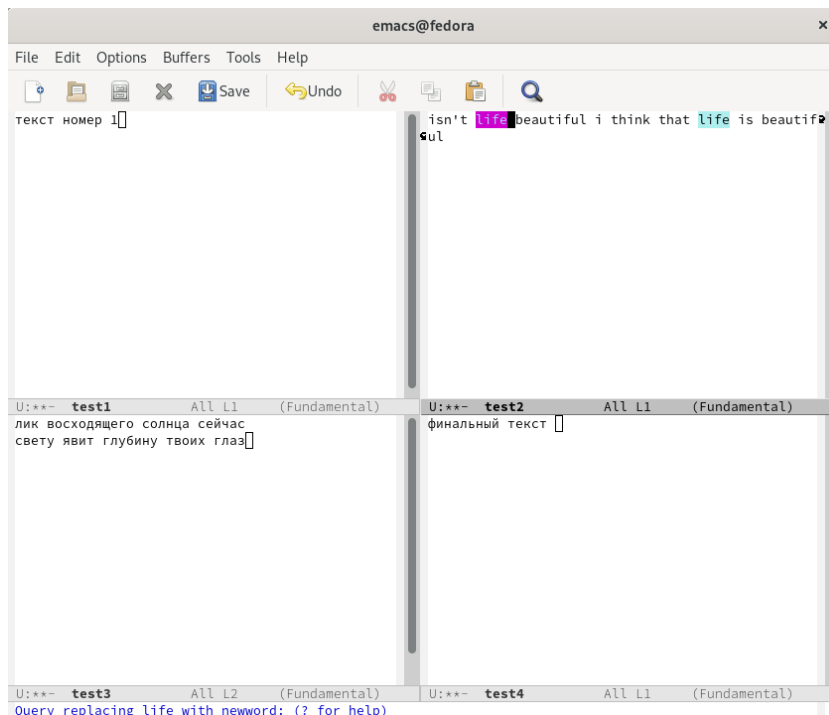


Рис. 0.31.: рис. 32

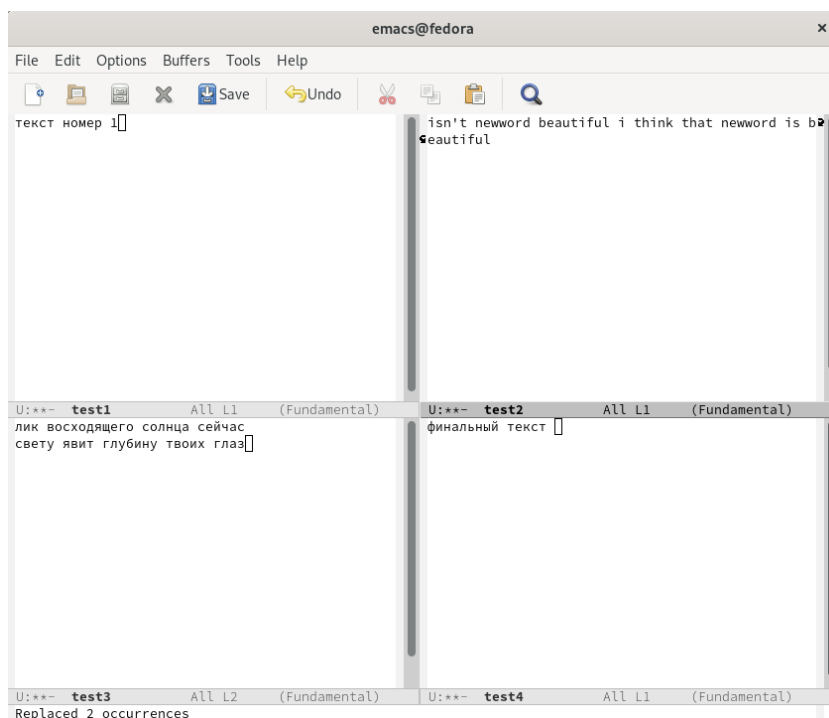


Рис. 0.32.: рис. 33

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

В обычном режиме нам просто подсвечивается текст в окне в буфере, а здесь у нас открывается результат поиска в отдельном окне в буфере * Occur * с ссылкой на буфер, где это слово было найдено. рис. [-@fig:034])

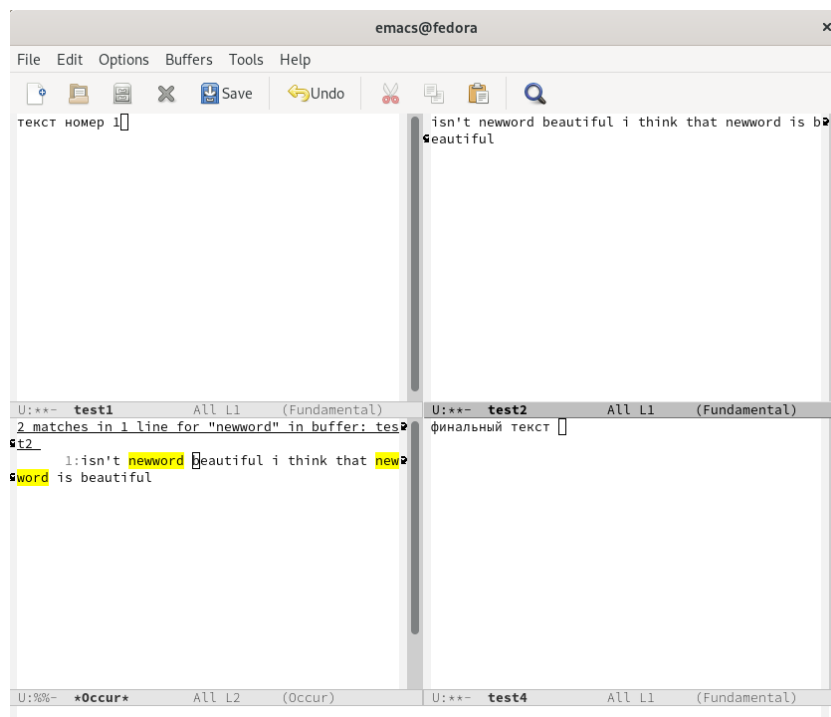


Рис. 0.33.: рис. 34

Выводы

В ходе работы я получил практические навыки работы с редактором Emacs.

Контрольные вопросы

1. Emacs – один из наиболее мощных и широко распространённых редакторов, используемых в мире Unix. По популярности он соперничает с редактором vi и его клонами. В зависимости от ситуации, Emacs может быть:
 - текстовым редактором;
 - программой для чтения почты и новостей Usenet;
 - интегрированной средой разработки (IDE);
 - операционной системой и т.д. Всё это разнообразие достигается благодаря архитектуре Emacs, которая позволяет расширять возможности редактора при помощи языка Emacs Lisp. На языке C написаны лишь самые базовые и низкоуровневые части Emacs, включая полнофункциональный интерпретатор языка Lisp. Таким образом, Emacs имеет встроенный язык программирования, который может использоваться для настройки, расширения и изменения поведения редактора. В действительности, большая часть того редактора, с которым пользователи Emacs работают в наши дни, написана на языке Lisp.
2. Основную трудность для новичков при освоении данного редактора могут составлять большое количество команд, комбинаций клавиш, которые не получится все запомнить с первого раза и поэтому придется часто обращаться к справочным материалам.
3. Буфер – это объект, представляющий собой текст. Если имеется несколько буферов, то редактировать можно только один. Обычно буфер считывает

данные из файла или записывает в файл данные из буфера. Окно – это область экрана, отображающая буфер. При запуске редактора отображается одно окно, но при обращении к некоторым функциям могут открыться дополнительные окна. Окна Emacs и окна графической среды X Window – разные вещи. Одно окно X Window может быть разбито на несколько окон в смысле Emacs, в каждом из которых отображается отдельный буфер.

4. Да, можно.
5. При запуске Emacs по умолчанию создаются следующие буферы:
 - «scratch» (буфер для несохраненного текста)
 - «Messages» (журнал ошибок, включающий также информацию, которая появляется в области EchoArea)
 - «GNU Emacs» (справочный буфер о редакторе)
6. C-c | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и нажимаю «|» C-c C-| сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и, удерживая «ctrl», нажимаю «|»
7. Чтобы поделить окно на две части необходимо воспользоваться комбинацией «Ctrl-x 3» (по вертикали) или «Ctrl-x 2» (по горизонтали).
8. Настройки Emacs хранятся в файле .emacs.
9. По умолчанию клавиша «Backspace» удаляет символ перед курсором, но в редакторе её можно переназначить. Для этого необходимо изменить конфигурацию файла .emacs.
10. Более удобным я считаю редактор emacs, потому что в нем проще открывать другие файлы, можно использовать сразу несколько окон, нет «Командного режима», «Режима ввода», «Режима командной строки», которые являются немного непривычными и в какой-то степени неудобными.