

Отчёт по лабораторной работе №9

Текстовый редактор emacs

Федюшина Ярослава Андреевна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	8
4	Выводы	33
5	Контрольные вопросы	34

Список иллюстраций

3.1	emacs	8
3.2	lab07.sh	9
3.3	текст задания	10
3.4	вырезаю строку	11
3.5	вставка в конец файла	12
3.6	выделение текста	13
3.7	вставка в конец файла	15
3.8	выделение и вырезка текста	16
3.9	отмена последнего действия	17
3.10	начало строки	18
3.11	конец строки	19
3.12	активные буферы	21
3.13	буферы	22
3.14	закрытие окна	23
3.15	по вертикали	25
3.16	по горизонтали	26
3.17	текст в окнах	27
3.18	режим поиска	28
3.19	переключаюсь в поиске	29
3.20	выход из поиска	30
3.21	режим поиска и замены	31

Список таблиц

1 Цель работы

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

2 Задание

1. Открыть emacs.
2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f(C-x C-f).
3. Наберите текст: `#!/bin/bash HELL=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) (см. рис. 9.1)
- 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?

3 Выполнение лабораторной работы

1. Открываю emacs

3.1

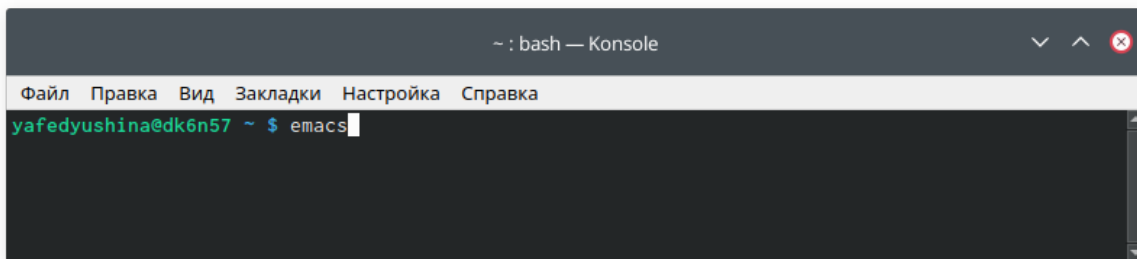


Рис. 3.1: emacs

3.2

2. Создаю файл lab07.sh с помощью комбинаций клавиш Ctrl-x и Ctrl-f.

3.3

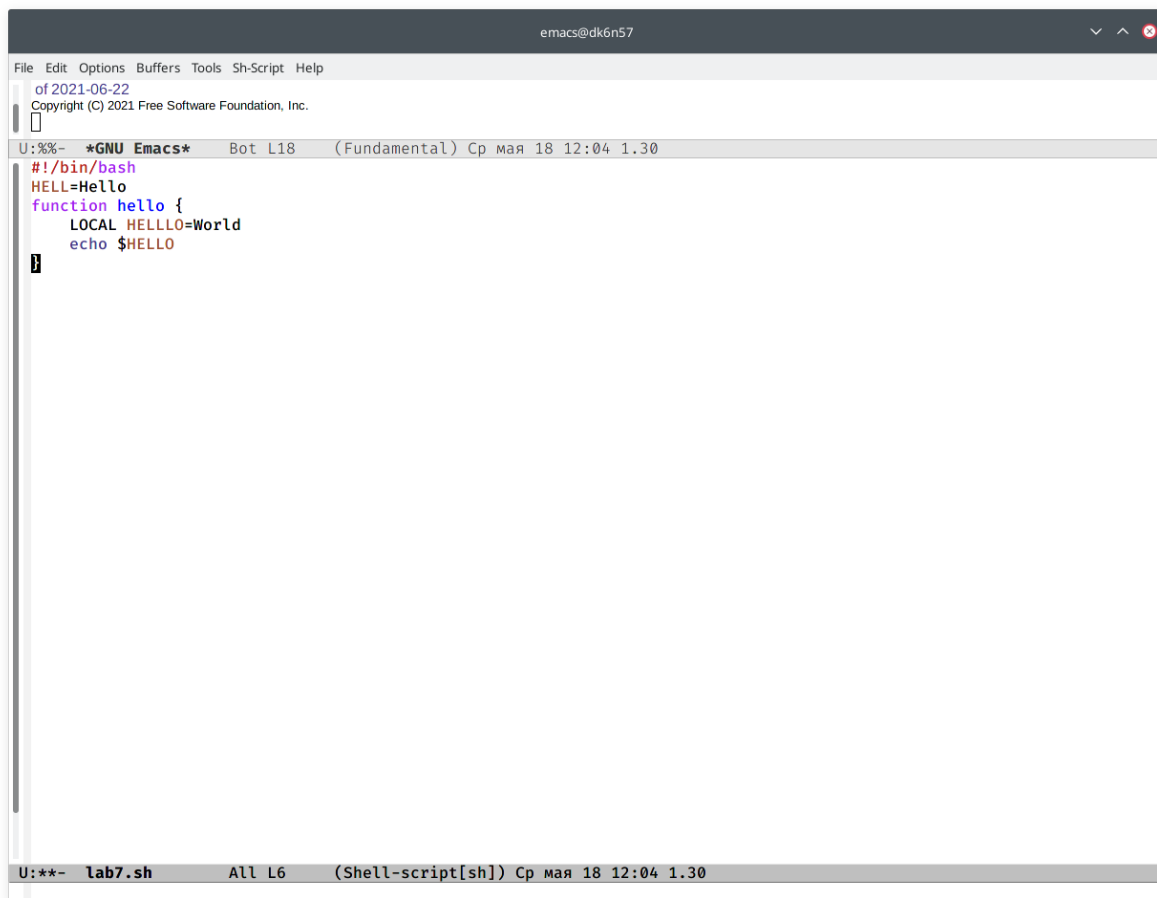


Рис. 3.2: lab07.sh

3.4

3. Набираю текст из задания:

3.5



The screenshot shows the Emacs editor window titled 'emacs@dk6n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main text area contains the following shell script:

```
of 2021-06-22
Copyright (C) 2021 Free Software Foundation, Inc.
[]

U:%%- *GNU Emacs* Bot L18 (Fundamental) Cp мая 18 12:04 1.30

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

The status bar at the bottom shows 'U:**~ lab7.sh All L6 (Shell-script[sh]) Cp мая 18 12:04 1.30'.

Рис. 3.3: текст задания

3.6

4. Сохраняю файл с помощью комбинации Ctrl-x и Ctrl-s.

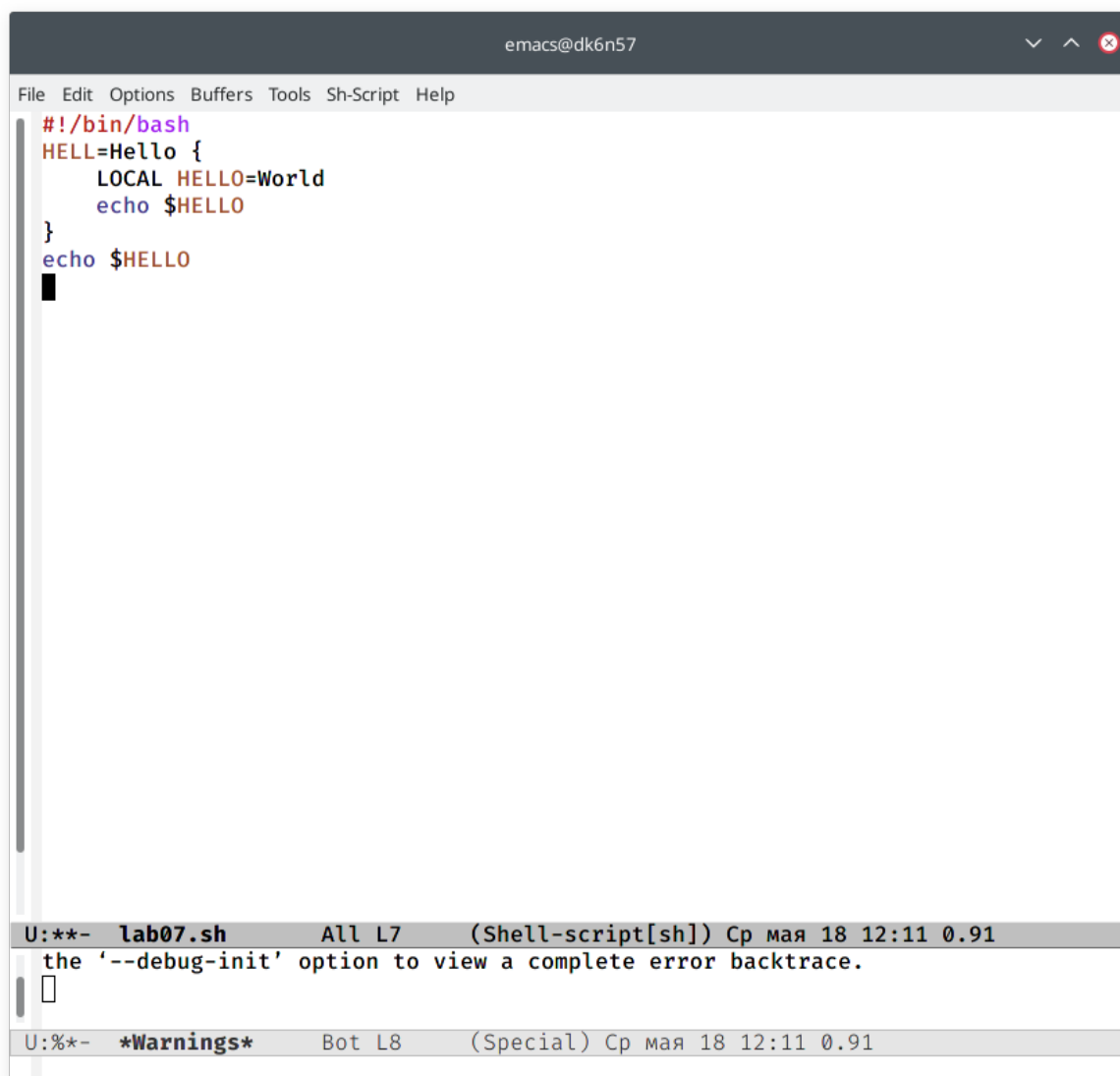
3.7

5. Проделываю с текстом стандартные процедуры редактирования:

3.8

- 5.1. Вырезаю одной командой целую строку (Ctrl-k)

3.9



The image shows an Emacs editor window titled 'emacs@dk6n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main editing area contains a shell script:

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

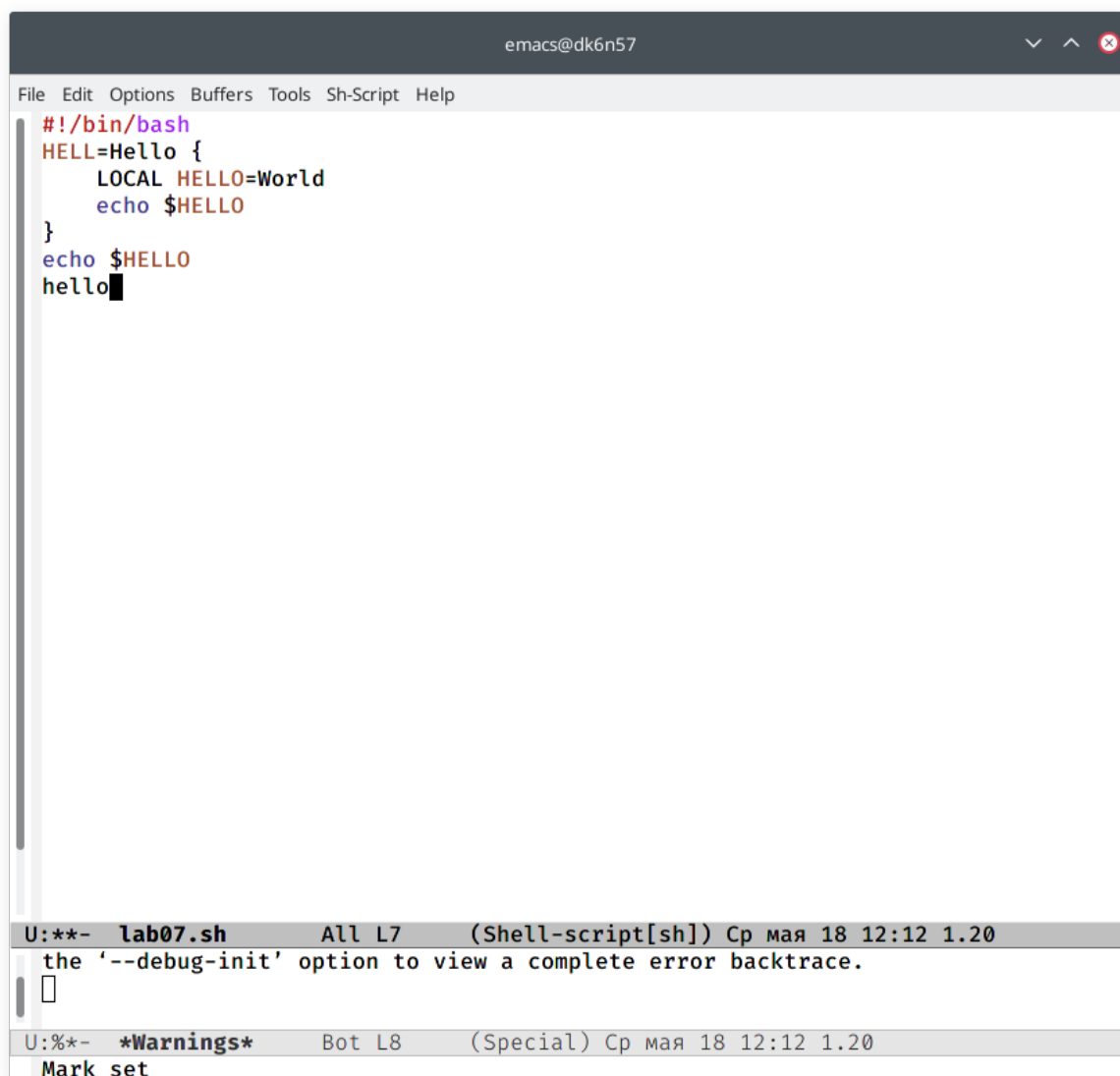
Below the editor, there are two terminal buffers. The first buffer shows a command prompt 'U:**-' followed by 'lab07.sh', 'All L7', '(Shell-script[sh])', and a timestamp 'Ср мая 18 12:11 0.91'. The second buffer shows a command prompt 'U:%*-' followed by '*Warnings*', 'Bot L8', '(Special)', and the same timestamp.

Рис. 3.4: вырезаю строку

3.10

- 5.2. Вставляю эту строку в конец файла (Ctrl-y)

3.11



```
emacs@dk6n57
File Edit Options Buffers Tools Sh-Script Help
#!/bin/bash
HELL=Hello {
  LOCAL HELLO=World
  echo $HELLO
}
echo $HELLO
hello
```

U:**~ lab07.sh All L7 (Shell-script[sh]) Ср мая 18 12:12 1.20
the '--debug-init' option to view a complete error backtrace.
[]

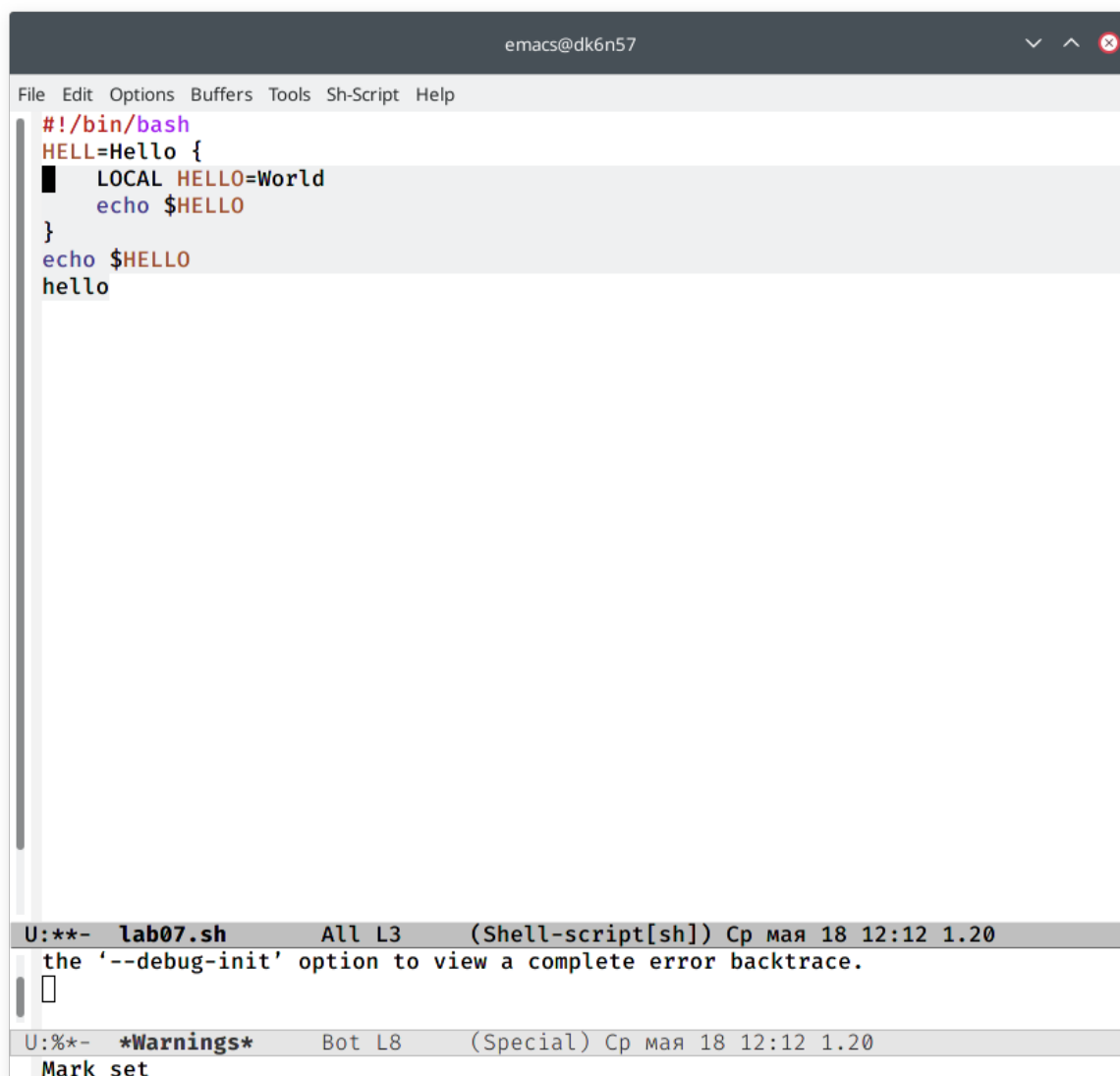
U:%*~ *Warnings* Bot L8 (Special) Ср мая 18 12:12 1.20
Mark set

Рис. 3.5: вставка в конец файла

3.12

- 5.3. Выделяю область текста (Ctrl-space)

3.13



The screenshot shows an Emacs editor window titled 'emacs@dk6n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main text area contains a shell script snippet:

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

A vertical bar on the left side of the text area indicates a text selection covering the lines from 'LOCAL HELLO=World' to 'echo \$HELLO'. The status bar at the bottom displays three lines of information:

- U:**~ lab07.sh All L3 (Shell-script[sh]) Cp мая 18 12:12 1.20
- the '--debug-init' option to view a complete error backtrace.
- U:%*~ *Warnings* Bot L8 (Special) Cp мая 18 12:12 1.20

The bottom line of the status bar also shows 'Mark set'.

Рис. 3.6: выделение текста

3.14

- 5.4. Копирую область в буфер обмена (meta-w)

3.15

- 5.5. Вставляю область в конец файла

3.16

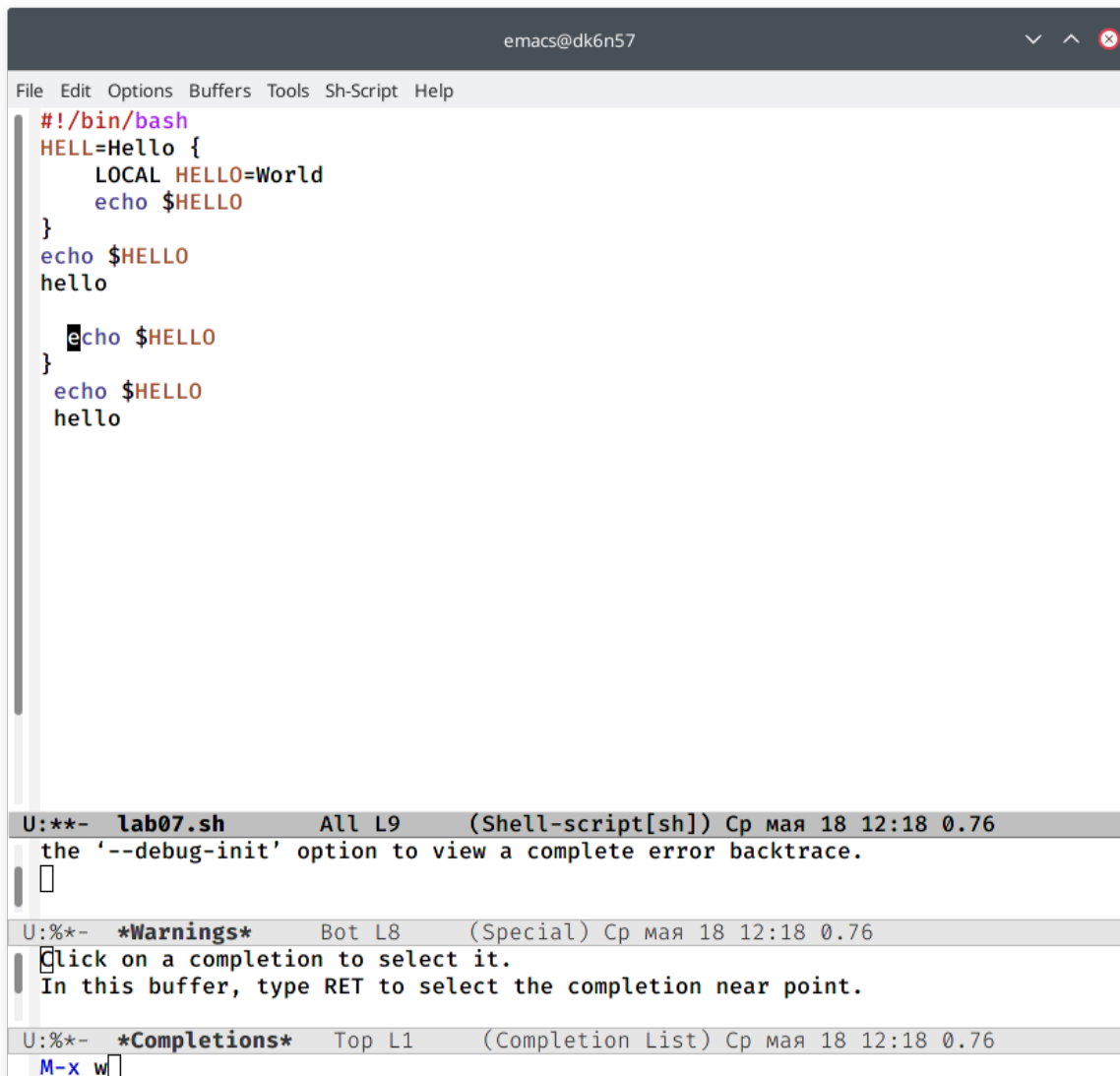
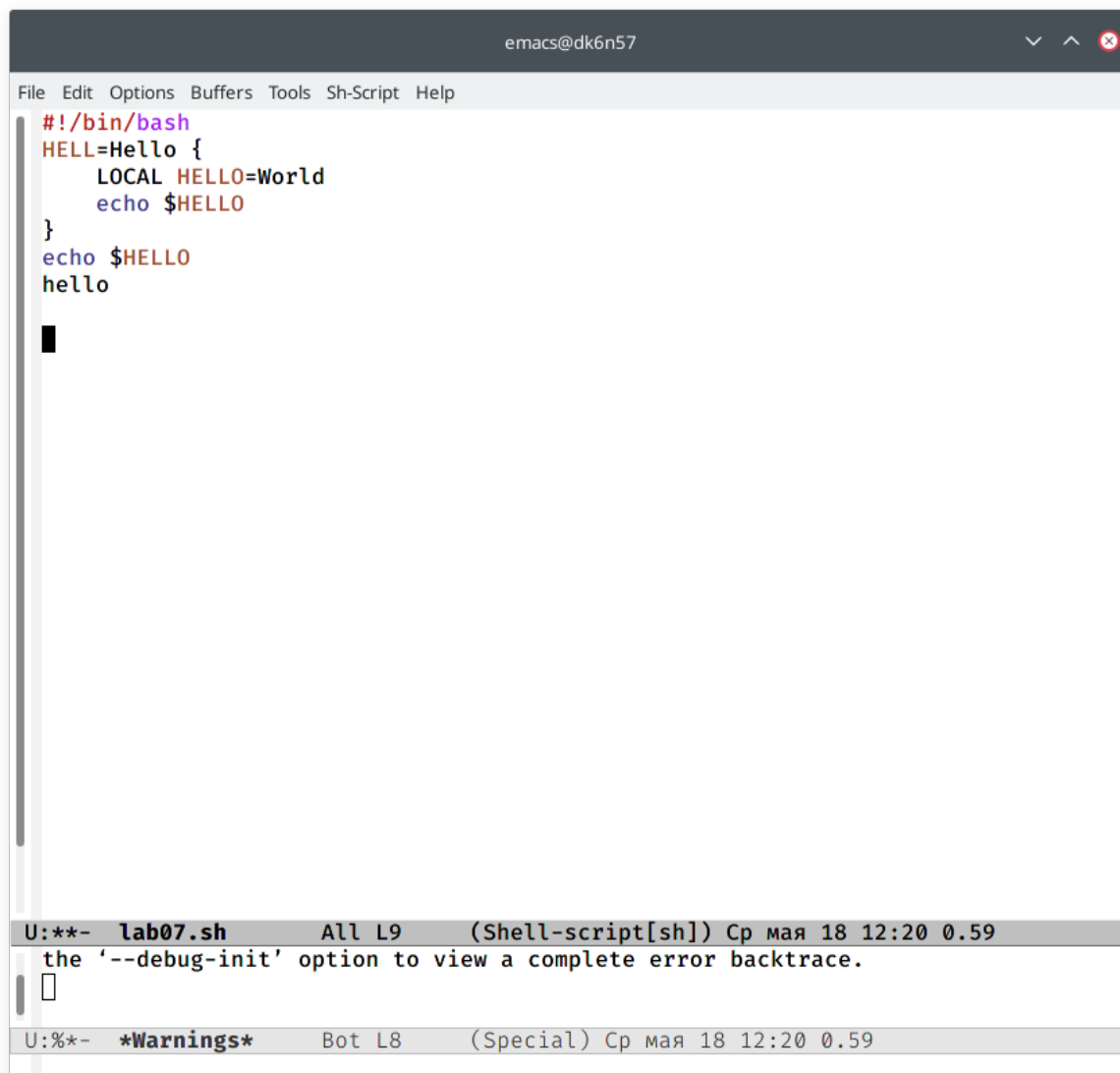


Рис. 3.7: вставка в конец файла

3.17

- 5.6. Снова выделяю эту область и вырезаю (Ctrl-w)

3.18



The screenshot shows an Emacs editor window titled 'emacs@dk6n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main editing area contains a shell script with the following content:

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

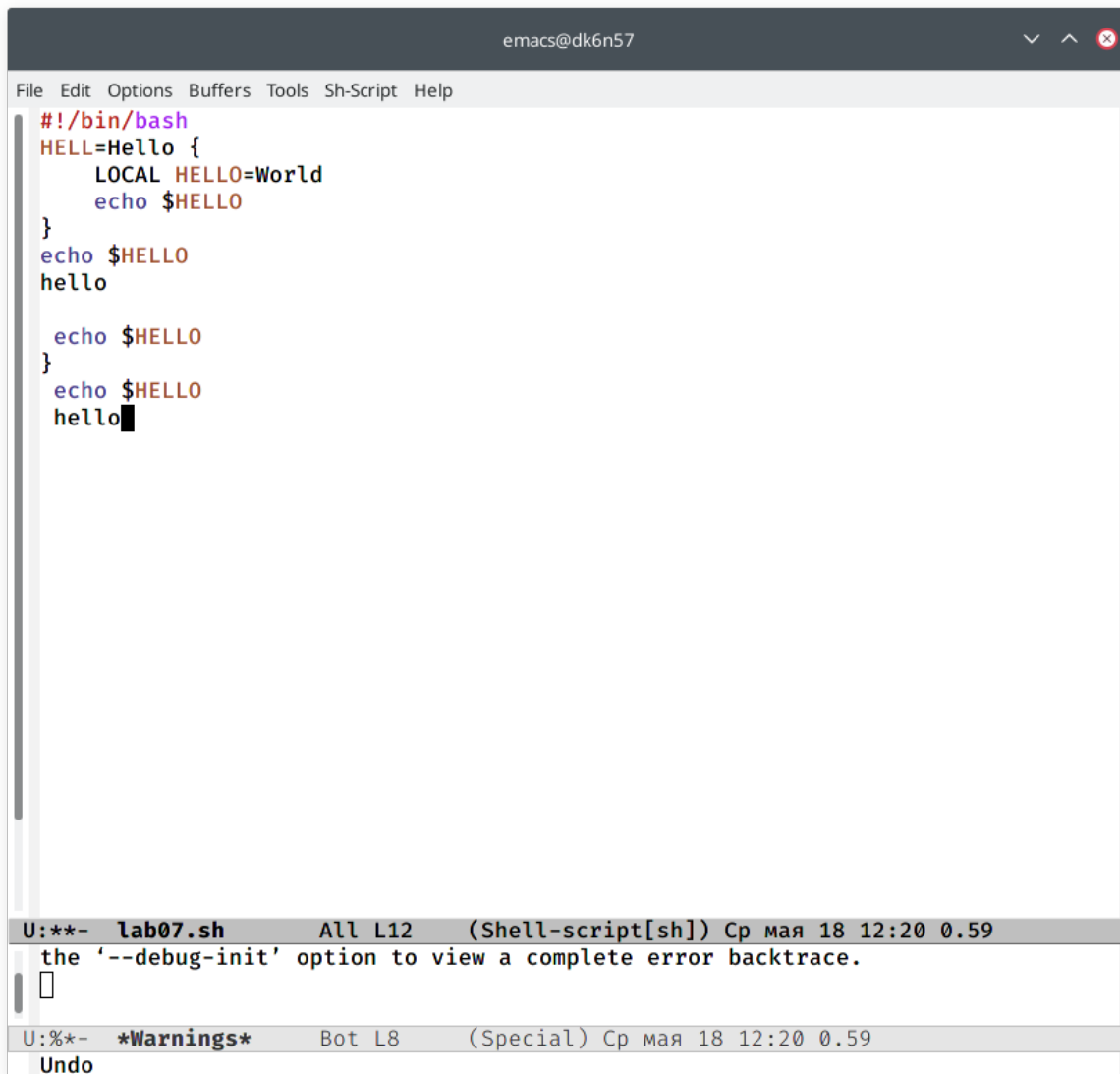
Below the editing area, there are two terminal buffers. The first buffer shows the command prompt 'U:***-' followed by 'lab07.sh', 'All L9', '(Shell-script[sh])', and 'Ср мая 18 12:20 0.59'. The second buffer shows the command prompt 'U:%*- ' followed by '*Warnings*', 'Bot L8', '(Special)', and 'Ср мая 18 12:20 0.59'.

Рис. 3.8: выделение и вырезка текста

3.19

- 5.7. Отменяю последнее действие (Ctrl - /)

3.20



The screenshot shows an Emacs editor window titled 'emacs@dk6n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main text area contains a shell script:

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

echo $HELLO
}
echo $HELLO
hello
```

Below the text area is a terminal buffer with two lines of output:

```
U:***- lab07.sh All L12 (Shell-script[sh]) Cp мая 18 12:20 0.59
the '--debug-init' option to view a complete error backtrace.
[]
U:%*- *Warnings* Bot L8 (Special) Cp мая 18 12:20 0.59
Undo
```

Рис. 3.9: отмена последнего действия

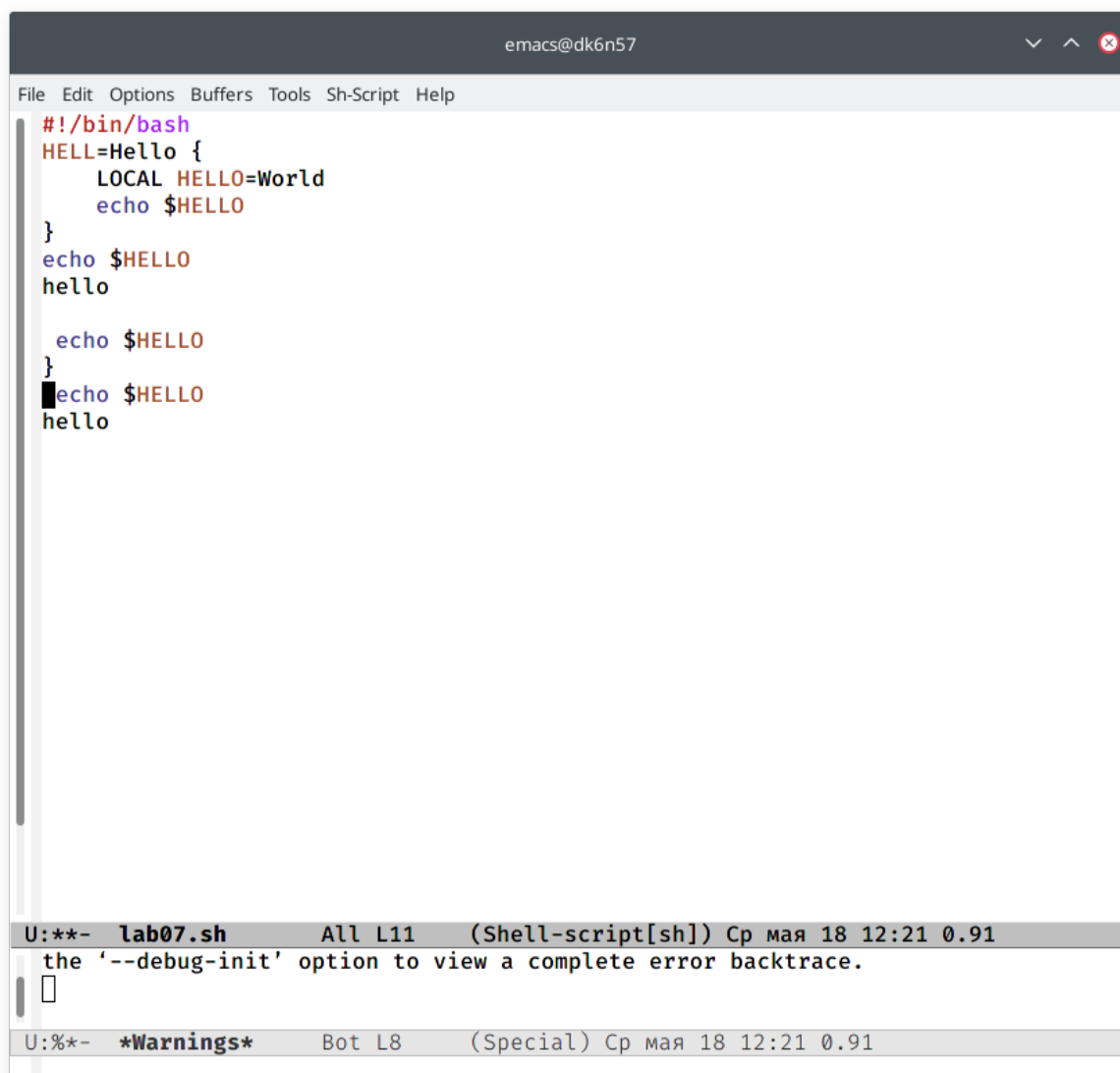
3.21

6. Учусь использовать команды по перемещению курсора

3.22

- 6.1. Перемещаю курсор в начало строки (ctrl-a)

3.23



The screenshot shows an Emacs editor window titled 'emacs@dk6n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main text area contains a shell script with the following content:

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

echo $HELLO
}
echo $HELLO
hello
```

The cursor is positioned at the beginning of the last line, 'hello'. Below the script, the output of the script is displayed in a terminal-like window. The output consists of three lines:

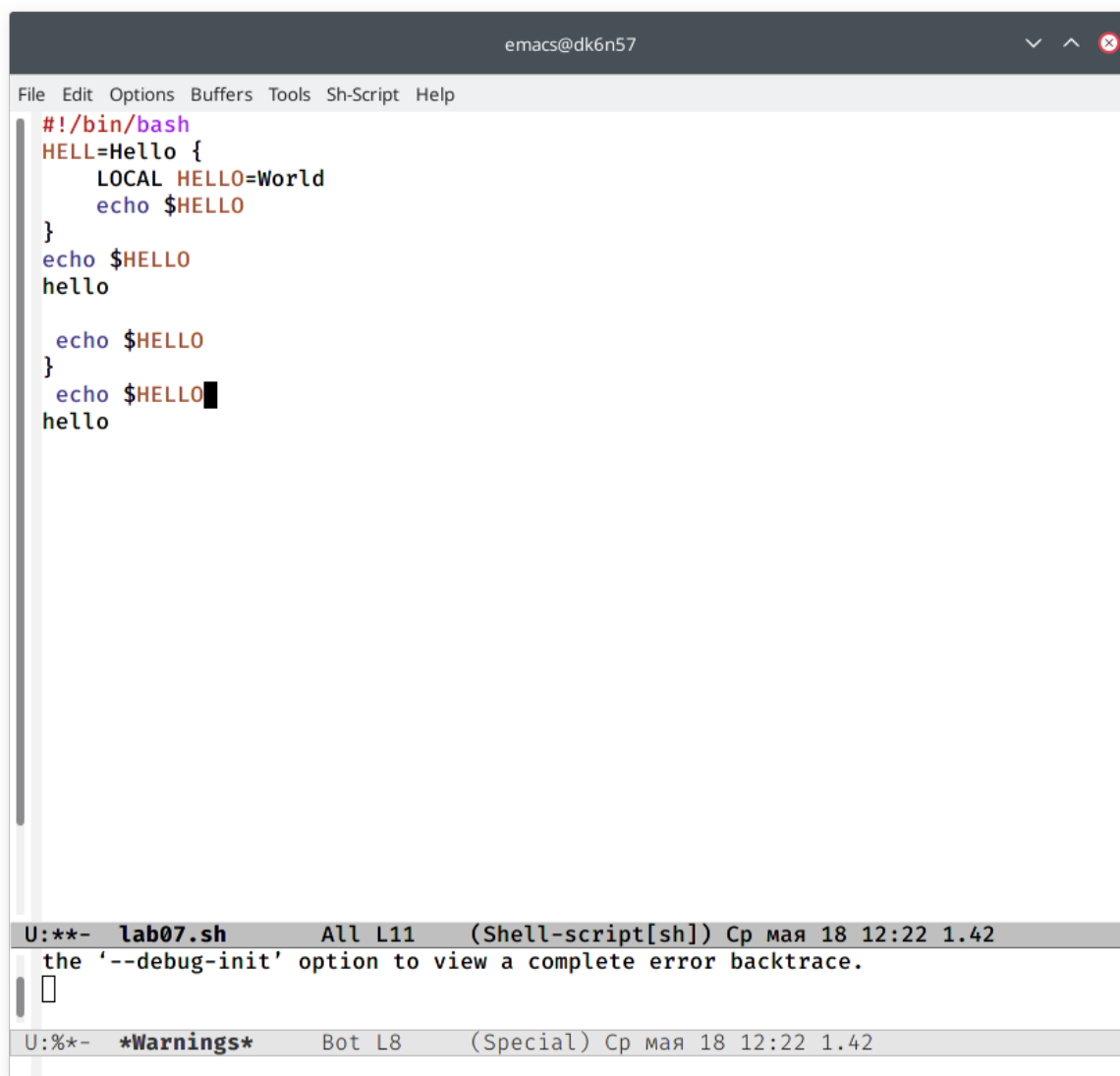
```
U:**~ lab07.sh All L11 (Shell-script[sh]) Cp мая 18 12:21 0.91
the '--debug-init' option to view a complete error backtrace.
U:%*~ *Warnings* Bot L8 (Special) Cp мая 18 12:21 0.91
```

Рис. 3.10: начало строки

3.24

- 6.2 Перемещаю курсор в конец строки (ctrl-e)

3.25



The screenshot shows an Emacs editor window titled 'emacs@dk6n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main editing area contains a shell script with the following content:

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

echo $HELLO
}
echo $HELLO
hello
```

The cursor is positioned at the end of the last line of the script. Below the editing area, there are two terminal buffers. The first buffer shows the command 'U:**~ lab07.sh' and the output 'All L11 (Shell-script[sh]) Cp мая 18 12:22 1.42 the '--debug-init' option to view a complete error backtrace.' The second buffer shows the command 'U:%*~ *Warnings*' and the output 'Bot L8 (Special) Cp мая 18 12:22 1.42'.

Рис. 3.11: конец строки

3.26

- 6.3 Перемещаю курсор в начало и конец буфера (meta-< / meta->)

3.27

7. Управление буферами

3.28

- 7.1 Вывожу список активных буферов на экран (ctrl-x и ctrl-b)

3.29

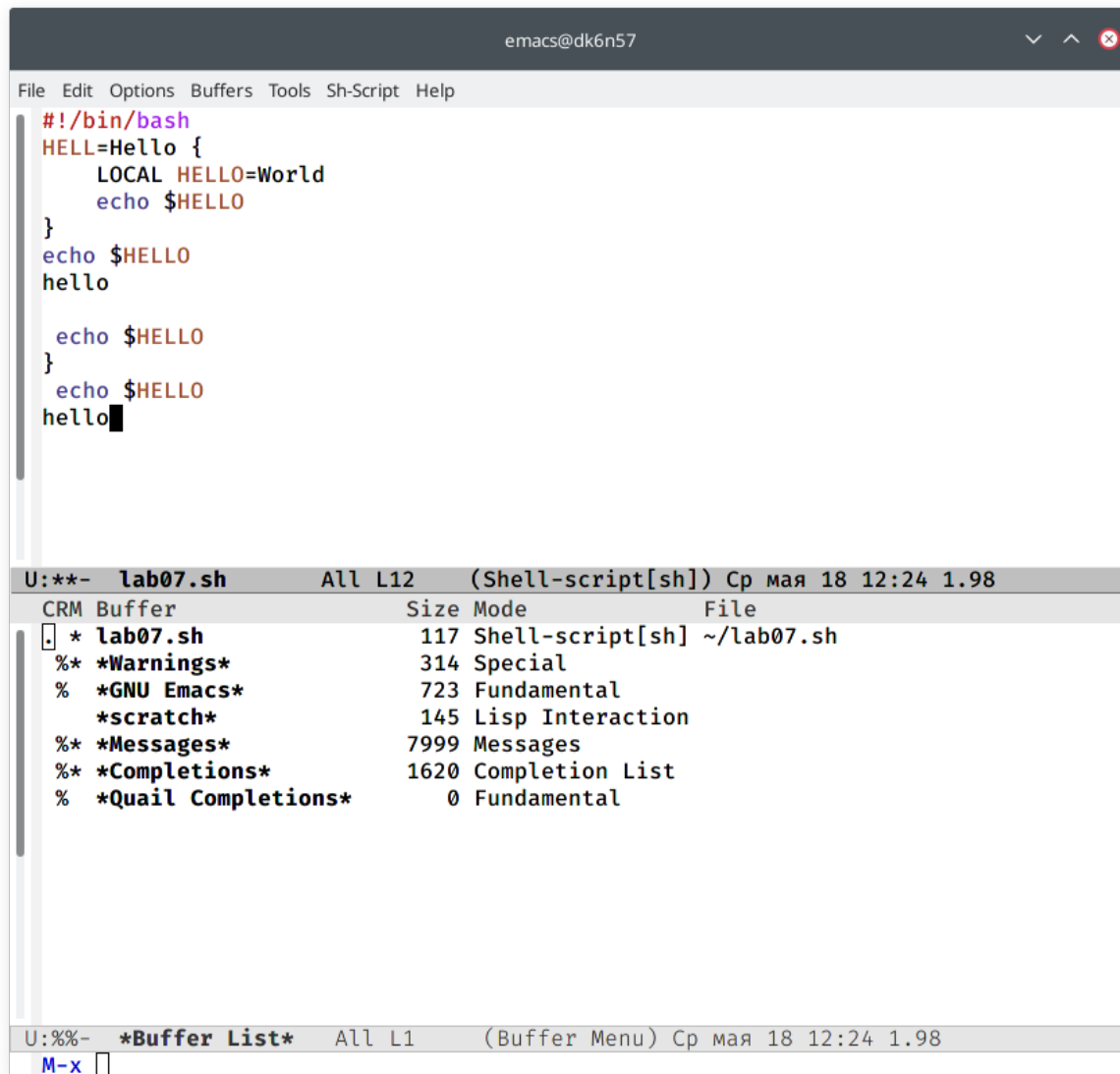


Рис. 3.12: активные буферы

3.30

- 7.2 Перемещаюсь во вновь открытое окно (ctrl-x) со списком открытых буферов и переключаюсь на другой буфер

3.31

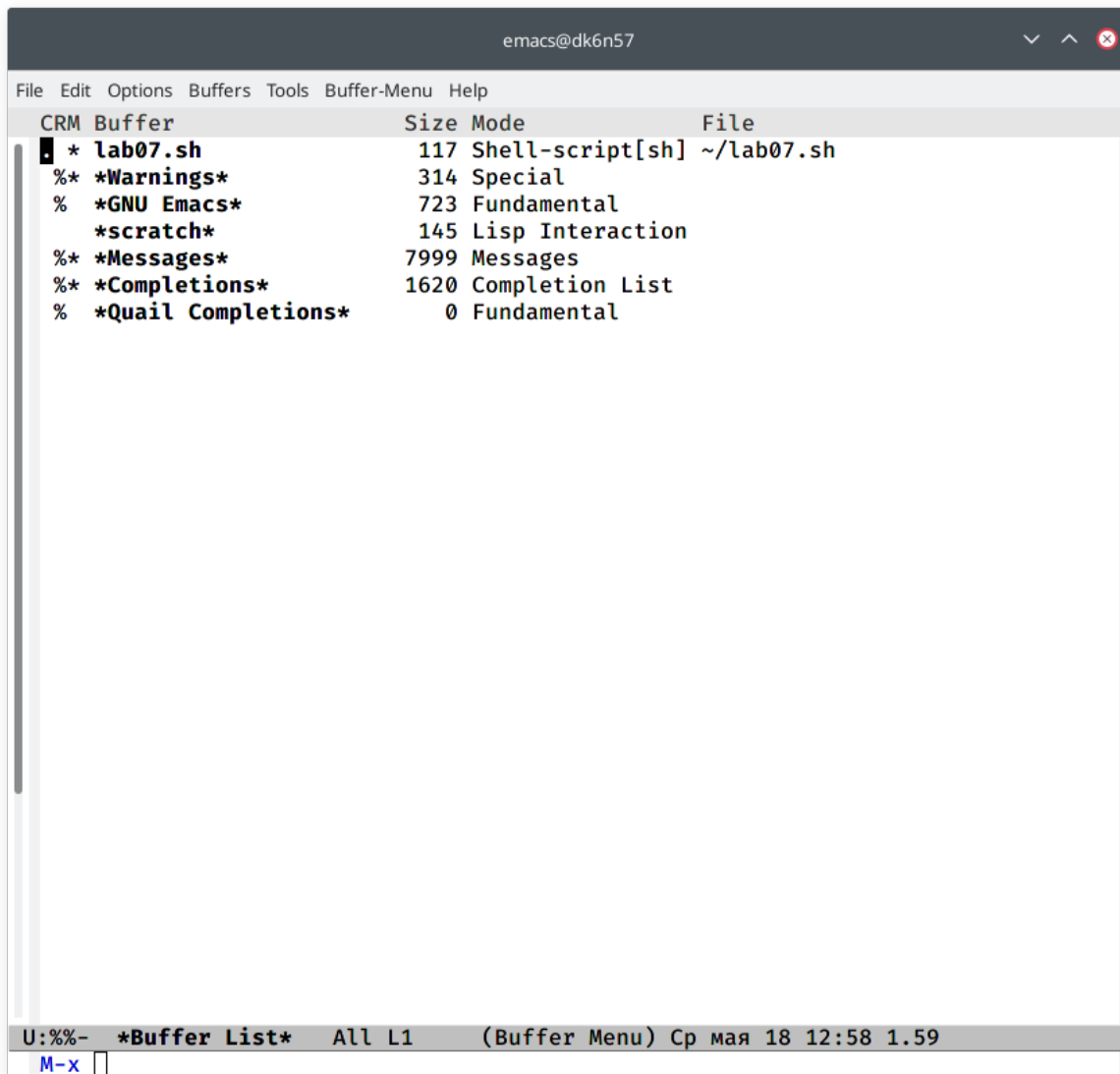
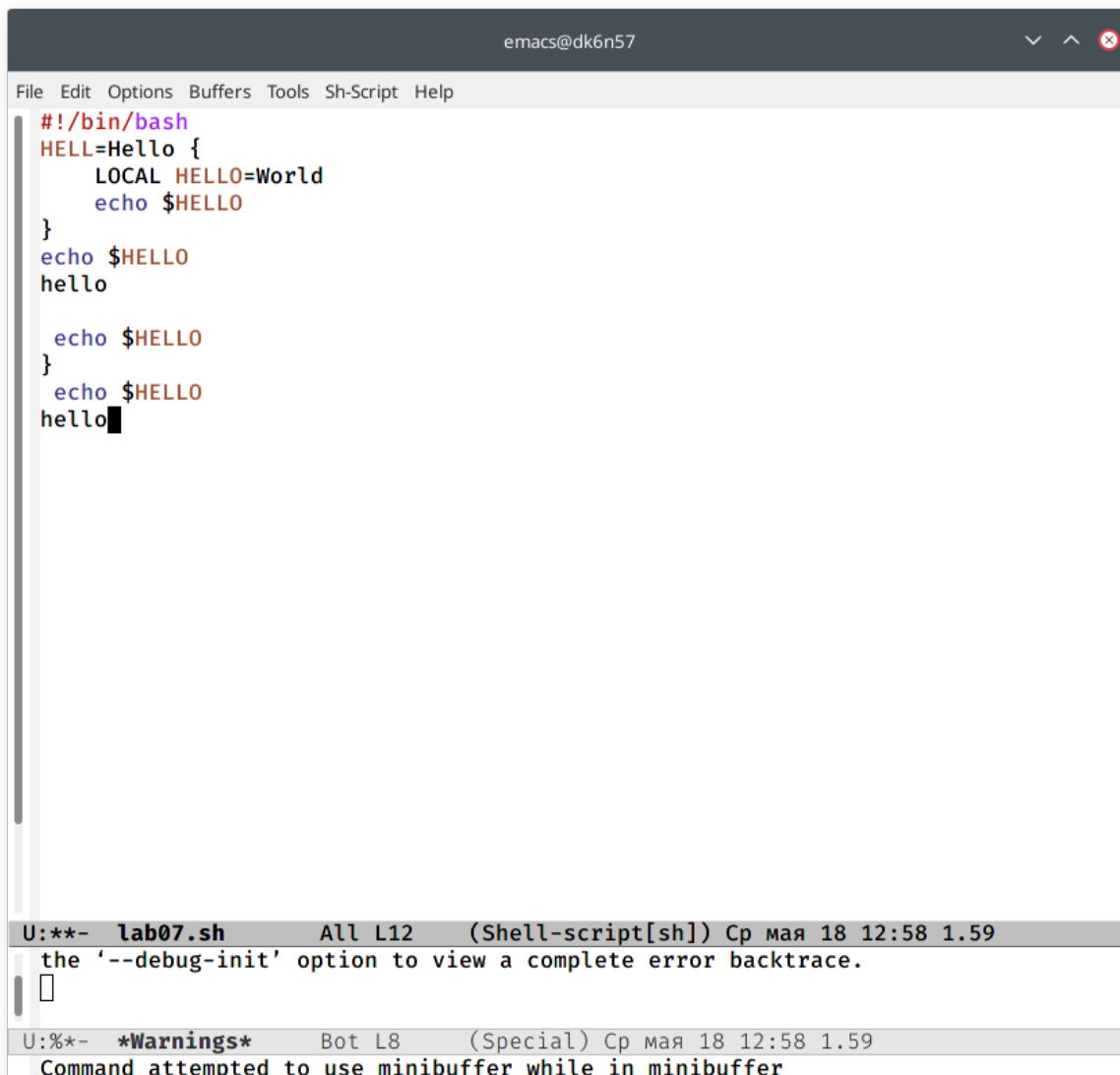


Рис. 3.13: буферы

3.32

- 7.3 Закрываю окно (ctrl-x 0)

3.33



The screenshot shows an Emacs window titled 'emacs@dk6n57'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The main text area contains a shell script:

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

echo $HELLO
}
echo $HELLO
hello
```

At the bottom, there is a terminal buffer with two lines of output:

```
U:***- lab07.sh All L12 (Shell-script[sh]) Ср мая 18 12:58 1.59
the '--debug-init' option to view a complete error backtrace.
[]
U:%*- *Warnings* Bot L8 (Special) Ср мая 18 12:58 1.59
Command attempted to use minibuffer while in minibuffer
```

Рис. 3.14: закрытие окна

3.34

- 7.4 Снова переключаюсь между буферами, но без вывода их списка на экран (ctrl-x b)

3.35

8. Управление окнами

3.36

- 8.1 Делю фрейм на 4 части: сначала по вертикали (ctrl-x 3), а затем каждое из этих окон на две части по горизонтали (ctrl-x 2)

3.37

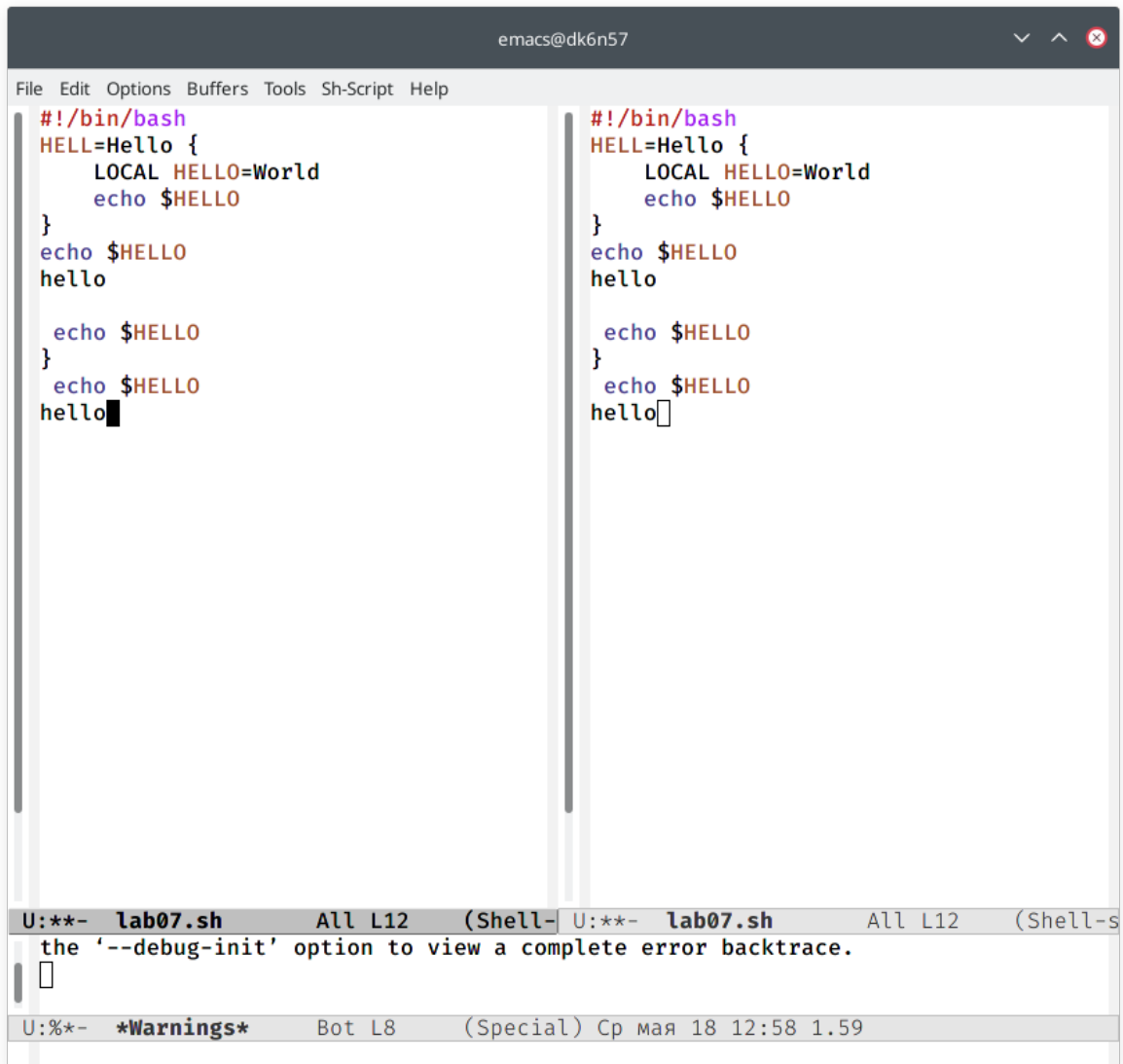


Рис. 3.15: по вертикали

3.38

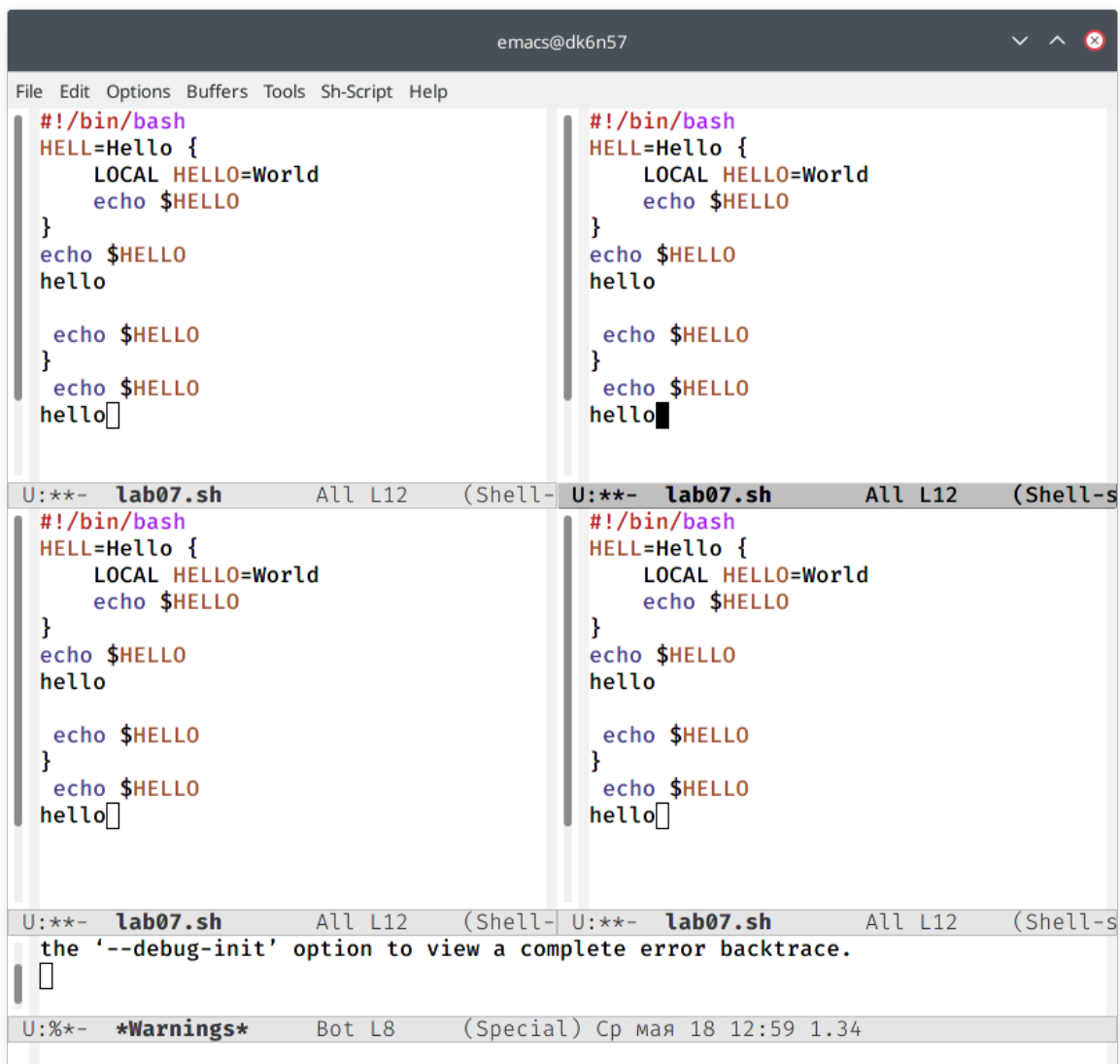


Рис. 3.16: по горизонтали

3.39

- 8.2 В каждом из четырёх окон открываю буфер и ввожу некоторый текст

3.40

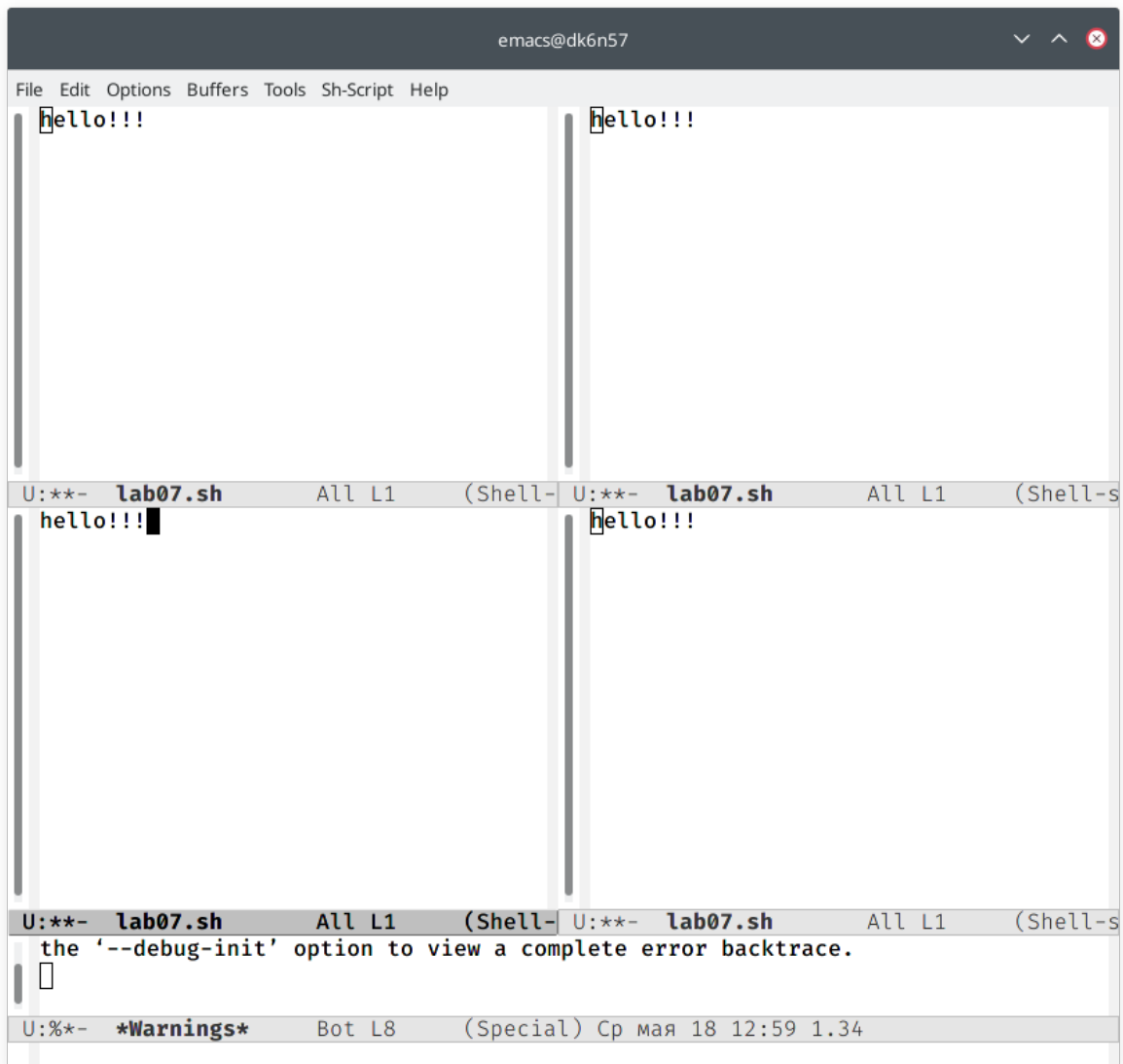


Рис. 3.17: текст в окнах

3.41

9. Режим поиска

3.42

- 9.1 Переключаюсь в режим поиска (ctrl-s) и ищу слово, которое имеется в тексте

3.43

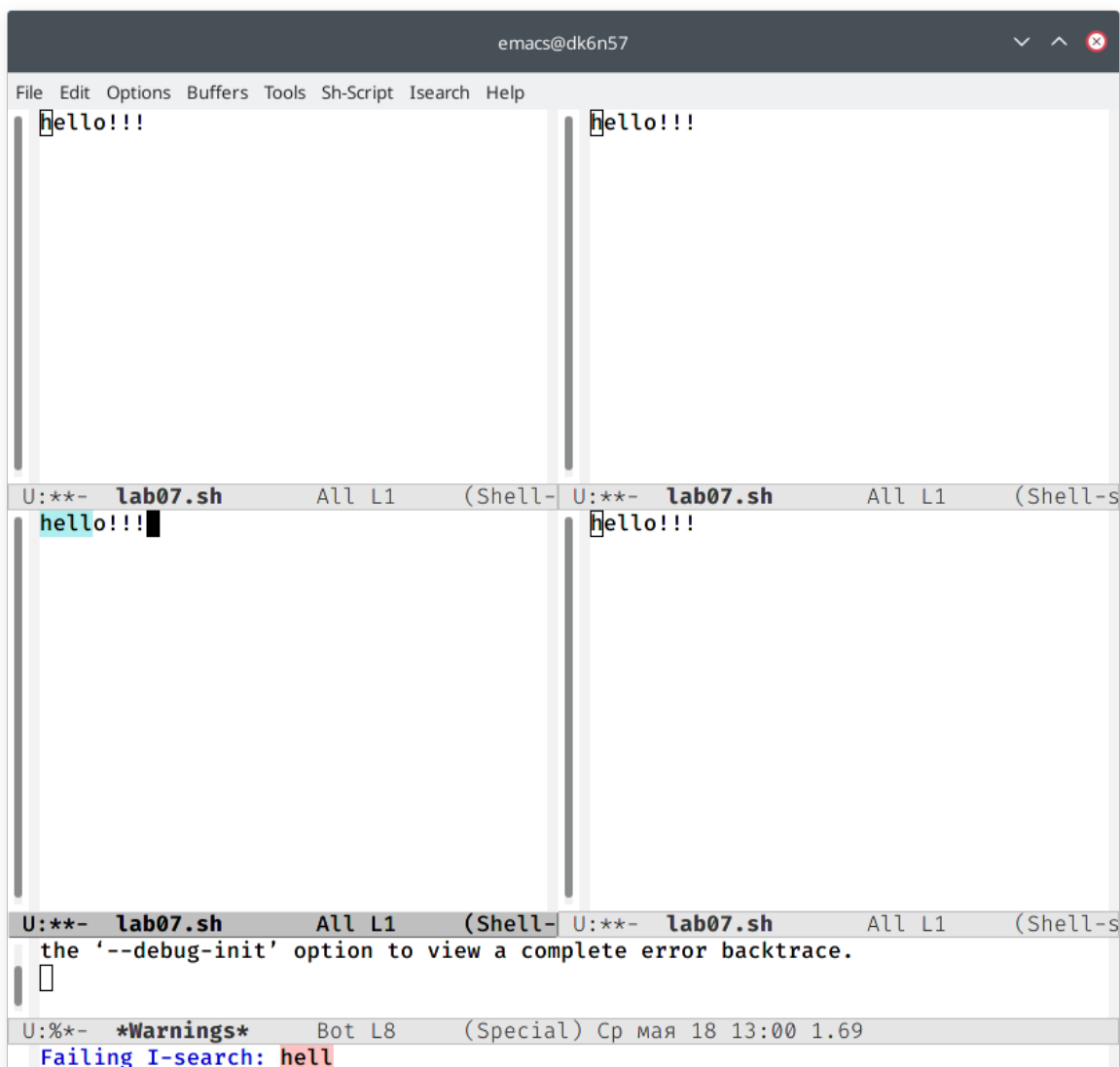


Рис. 3.18: режим поиска

3.44

- 9.2 Переключаюсь между результатами поиска (ctrl-s)

3.45

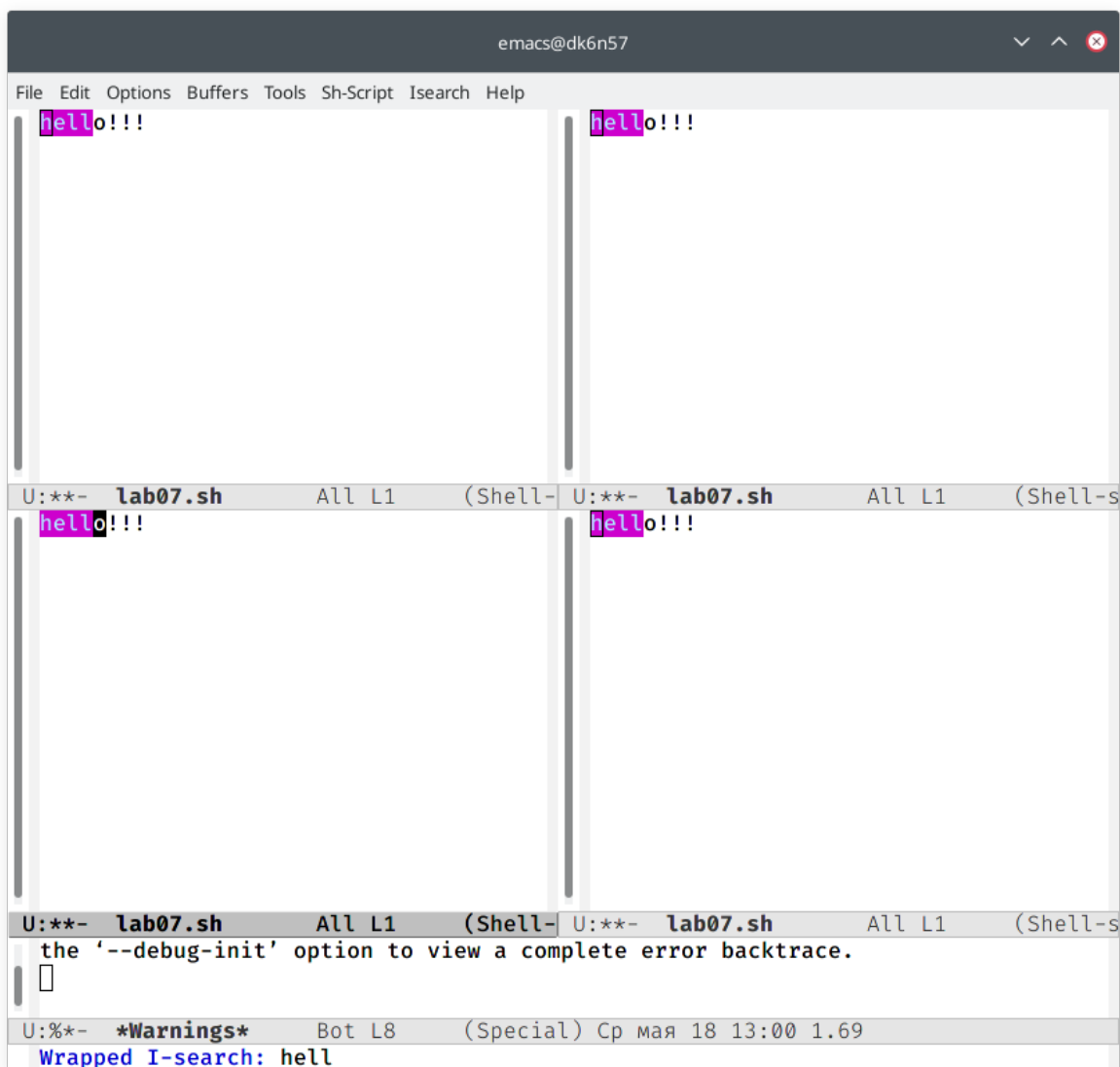


Рис. 3.19: переключаюсь в поиске

3.46

- 9.3 Выхожу из режима поиска (ctrl-g)

3.47

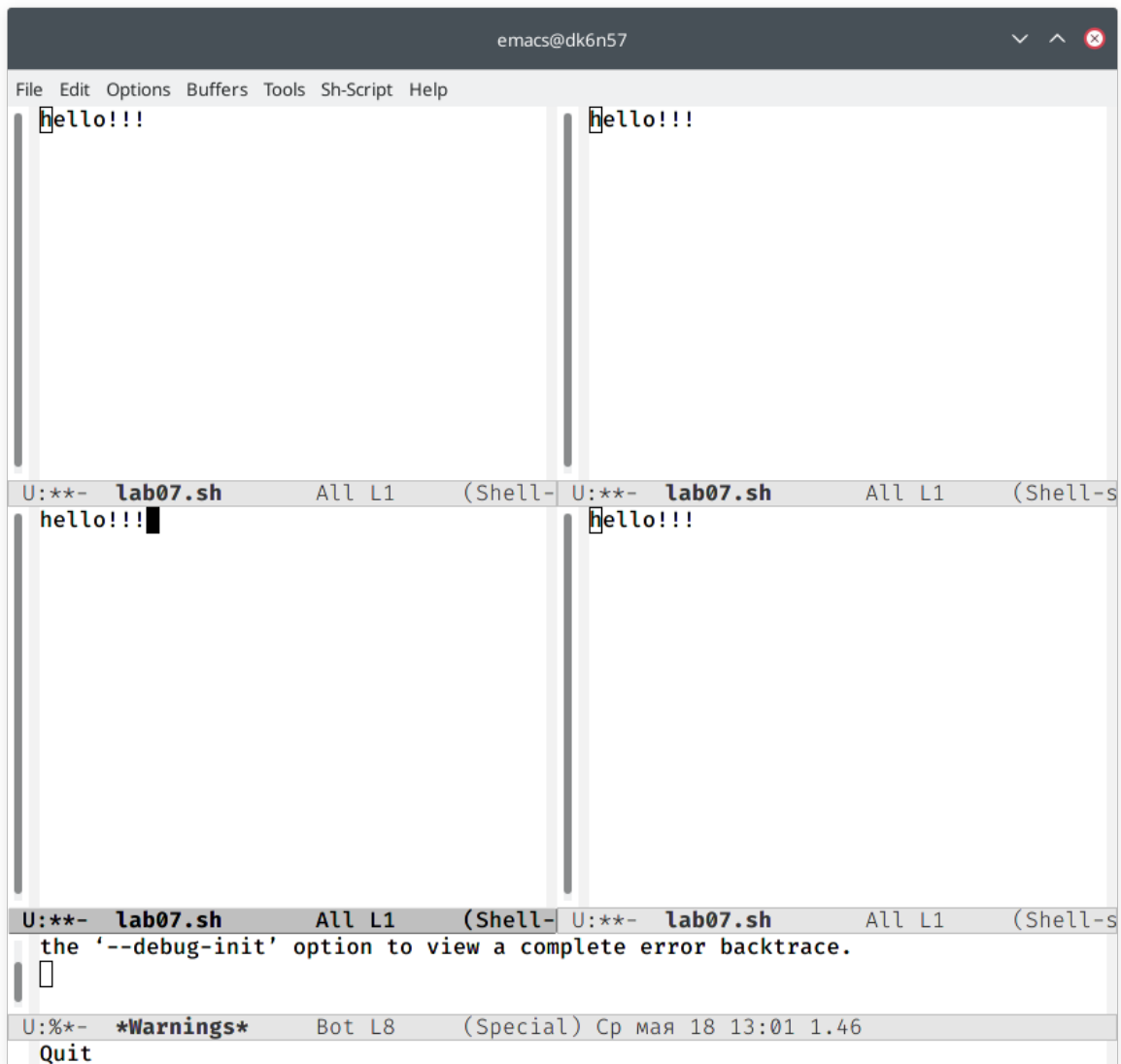


Рис. 3.20: выход из поиска

3.48

- 9.4 Перехожу в режим поиска и замены (meta-%) и ввожу текст, который следует найти и заменить

3.49

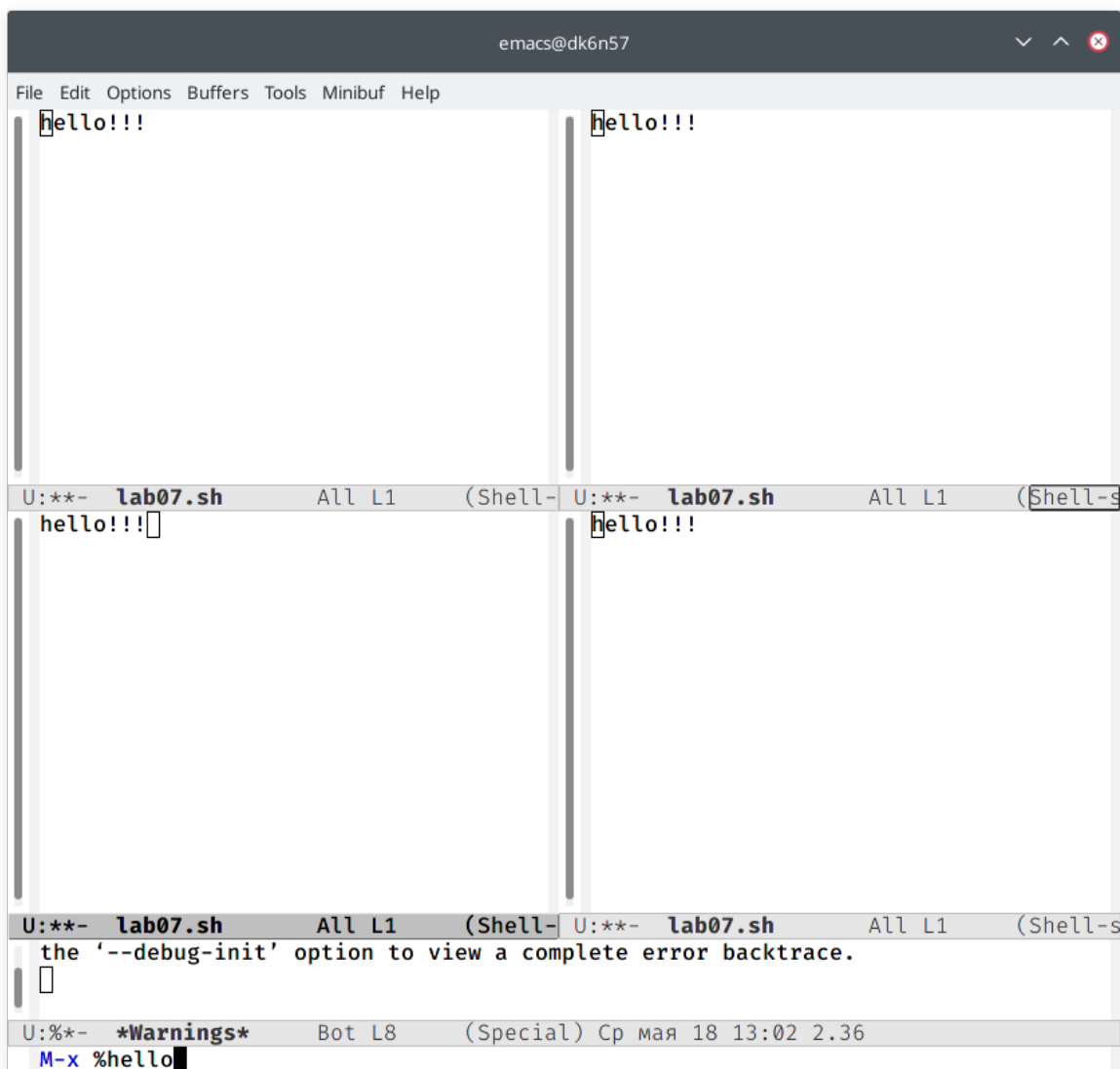


Рис. 3.21: режим поиска и замены

3.50

4 Выводы

В ходе выполнения лабораторной работы №9 я познакомилась с операционной системой Linux и получила практические навыки работы с редактором emacs

5 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.