

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 10

дисциплина: Операционные системы

Студент:

Афтаева Ксения Васильевна

Преподаватель:

Велиева Т.В.

Группа: НПИбд-01-20

МОСКВА 2021 г.

Цель работы:

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

Задачи:

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

Объект и предмет исследования:

ОС Linux, редактор Emacs

Техническое оснащение:

Ноутбук, на котором установлена виртуальная машина с линукс и редактор Emacs

Теоретические вводные данные [1] :

В то время как vi является без сомнения самым распространённым редактором в Unix-подобных системах, **Emacs** считается вторым хорошим редактором. Вместо

различных “режимов”, применяемых в vi, для ввода команд в нём используются комбинации клавиш с Control и Alt.

Также, в отличие от vi, **Emacs** - это программа с практически неограниченными возможностями. **Emacs** (по большей части) написан на *Lisp* - очень мощном языке программирования, характерной особенностью которого является то, что любая написанная на этом языке программа автоматически сама для себя является компилятором Lisp. Это означает, что пользователь может самостоятельно расширять возможности **Emacs**, а по сути писать новые программы “на Emacs'e”.

Каждый *буфер* в **Emacs** имеет связанный с ним режим. Этот режим очень отличается от идеи режимов в vi: режим говорит вам о типе буфера, в котором вы находитесь. Например, для обычных текстовых файлов существует *text-mode* (текстовый режим), однако также существуют такие режимы как: *c-mode* (режим C) для редактирования программ на C, *sh-mode* (режим командного процессора) для редактирования шелл-скриптов, *latex-mode* (режим latex) для редактирования файлов LaTeX, *mail-mode* (почтовый режим) для редактирования электронных писем и новостей и другие режимы. Режимы предоставляют специальные возможности и функции, полезные для различных типов редактируемых файлов. Для режимов даже можно переопределять клавиши и клавишные команды. Например, в режиме текста клавиша Tab выполняет перескакивание на окончание следующего табулятора, а в большинстве режимов для языков программирования клавиша Tab делает отступ текущей строки на соответствующую глубину текущего блока, в котором находится эта строка.

Упомянутые выше режимы относятся к главным режимам. У каждого буфера есть только один главный режим. Дополнительно буфер может иметь один или несколько второстепенных режимов. Второстепенный режим предоставляет дополнительные возможности, которые могут быть полезными в определённых задачах редактирования. Например, если вы нажмёте клавишу INSERT, вы включите режим *замены*, т.е. именно то, что вы и ожидали от этой клавиши. Есть ещё режим *автозаполнения* (auto-fill), который удобен в комбинации с режимами текста или latex: в нём каждая набранная вами строка будет автоматически разбиваться с переносом на следующую строку при достижении определённого количества символов.

Условные обозначения и символы:

C - клавиша ctrl

M - клавиша meta (alt)

S - клавиша shift

Выполнение работы:

1. Ознакомилась с теоретическим материалом, представленным в описании к лабораторной работе №10 [2]
2. Установила текстовый редактор **emacs**. Для этого я сначала перешла в режим суперпользователя с помощью команды `su` и введения пароля. После этого ввела команду `yum install emacs` для установки (Рис.1)

```
[kvaftaeva@kvaftaeva ~]$ su
Пароль:
[root@kvaftaeva kvaftaeva]# yum install emacs
```

Рис.1 Начало установки

Видим, что установка прошла успешно по соответствующей надписи на экране. Перехожу из режима суперпользователя обратно с помощью команды `su kvaftaeva` (Рис.2)

```
Установлено:
  emacs.x86_64 1:24.3-23.el7

Установлены зависимости:
  ImageMagick.x86_64 0:6.9.10.68-5.el7_9      OpenEXR-libs.x86_64 0:1.7.1-8.el7
  emacs-common.x86_64 1:24.3-23.el7           ilmbase.x86_64 0:1.0.3-7.el7
  libXaw.x86_64 0:1.0.13-4.el7                 libblockfile.x86_64 0:1.08-17.el7
  libotf.x86_64 0:0.9.13-4.el7                 libwmf-lite.x86_64 0:0.2.8.4-44.el7

Выполнено!
[root@kvaftaeva kvaftaeva]# su kvaftaeva
[kvaftaeva@kvaftaeva ~]$
```

Рис.2 Завершение установки

3. Открываю emacs, введя в командной строке `emacs` (Рис.3)

```
[kvaftaeva@kvaftaeva ~]$ emacs
```

Рис.3 Открытие emacs

Видим, что редактор открылся в отдельном окне (Рис.4)

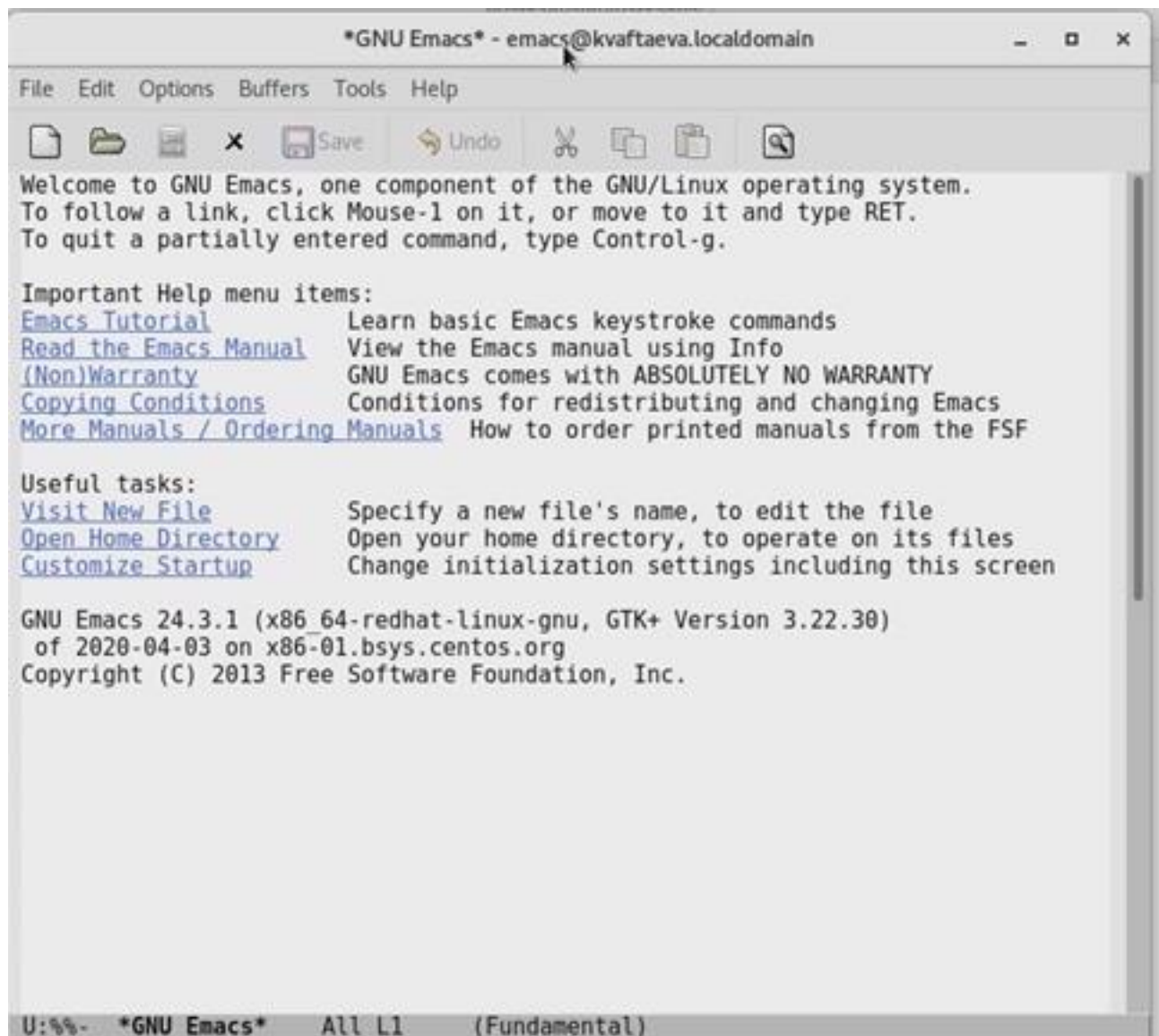


Рис.4 Редактор emacs

4. Нам предлагается создать файл **lab07.sh**. Однако, это название совпадает со старой нумерацией лабораторных работ. Поэтому я назову файл в соответствии с новой нумерацией - **lab10.sh**. Создадим его с помощью комбинации C-x C-f. Появляется фрейм, в котором нам предлагают ввести имя редактируемого файла, что я и делаю (Рис.5)



Рис.5 Создание файла

Однако так как такого файла пока нет, он автоматически создается данной командой, а потом открывается на редактирование (Рис.6)

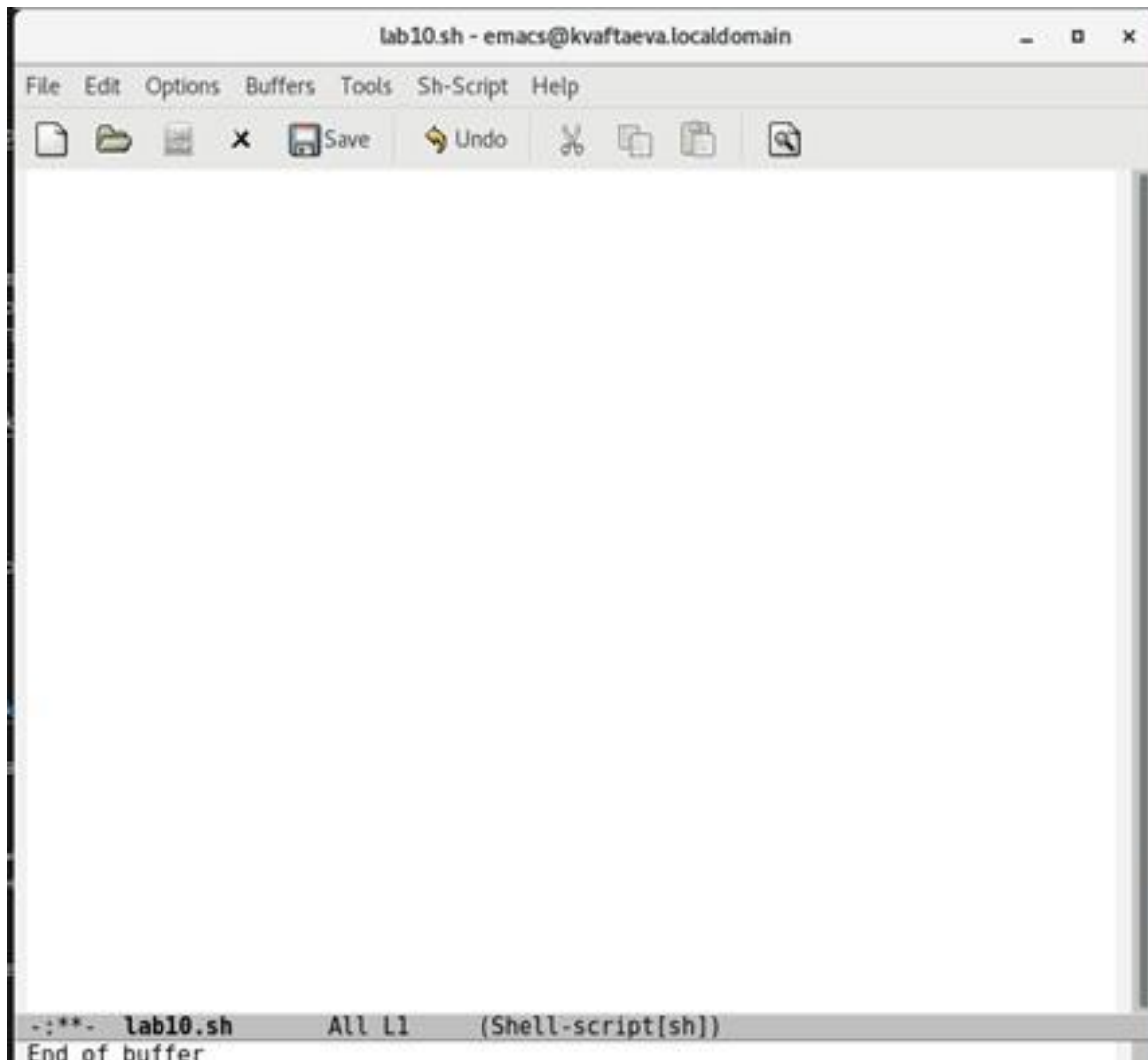


Рис.6 Открытие файла

5. Набираю предложенный в описании лабораторной работы (Рис.7)

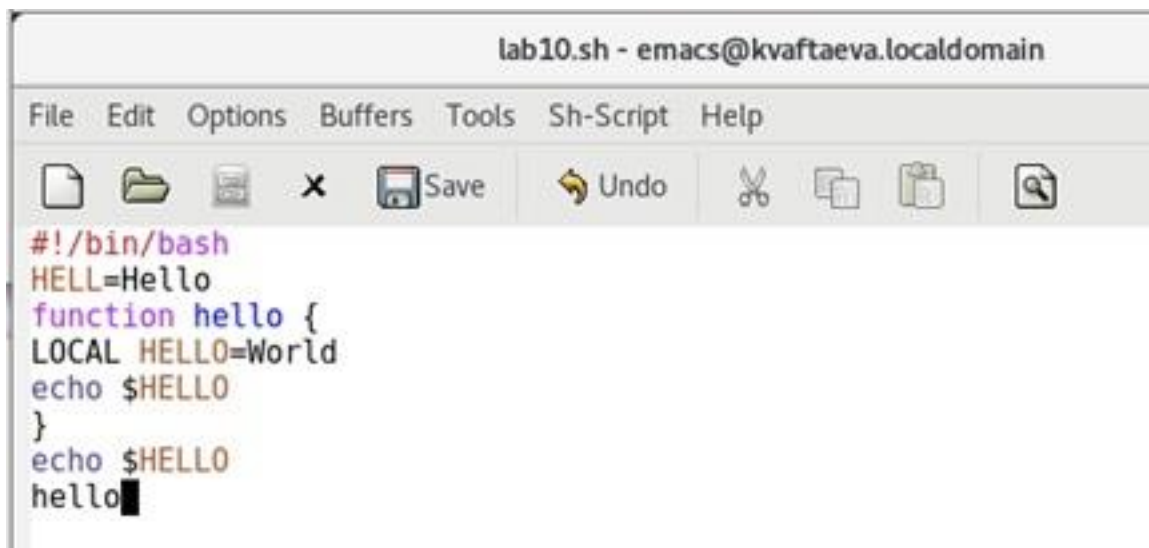


Рис.7 Ввод текста

6. Сохранила файл с помощью комбинации клавиш C-x C-s (Рис.8)

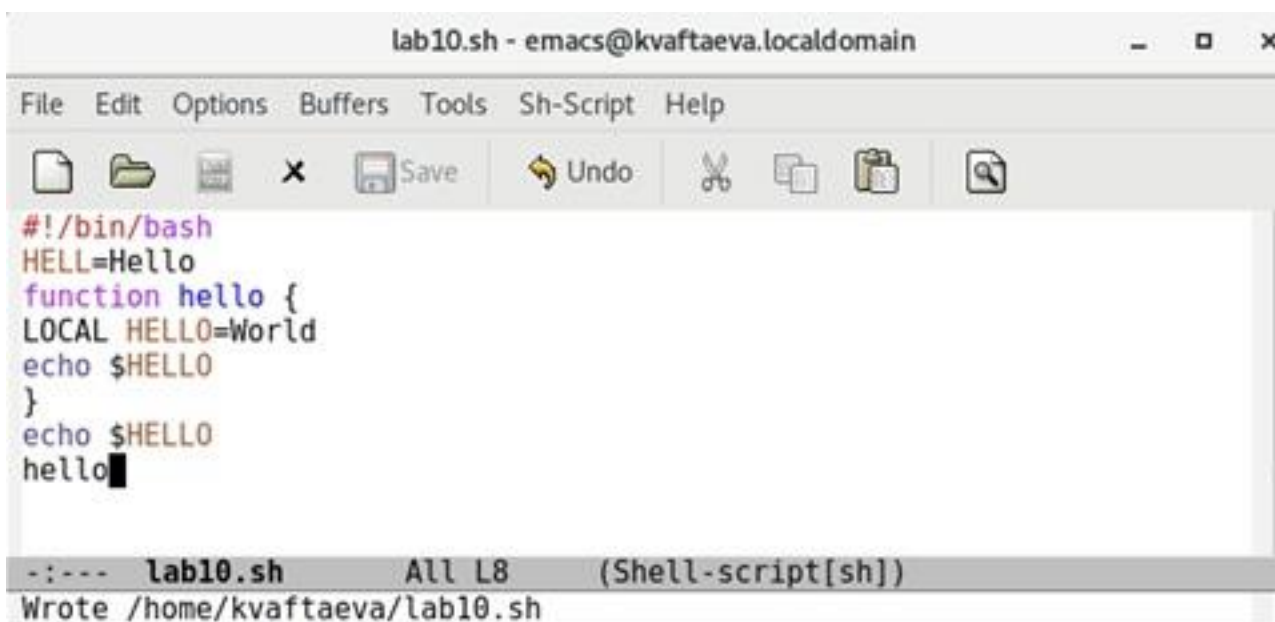
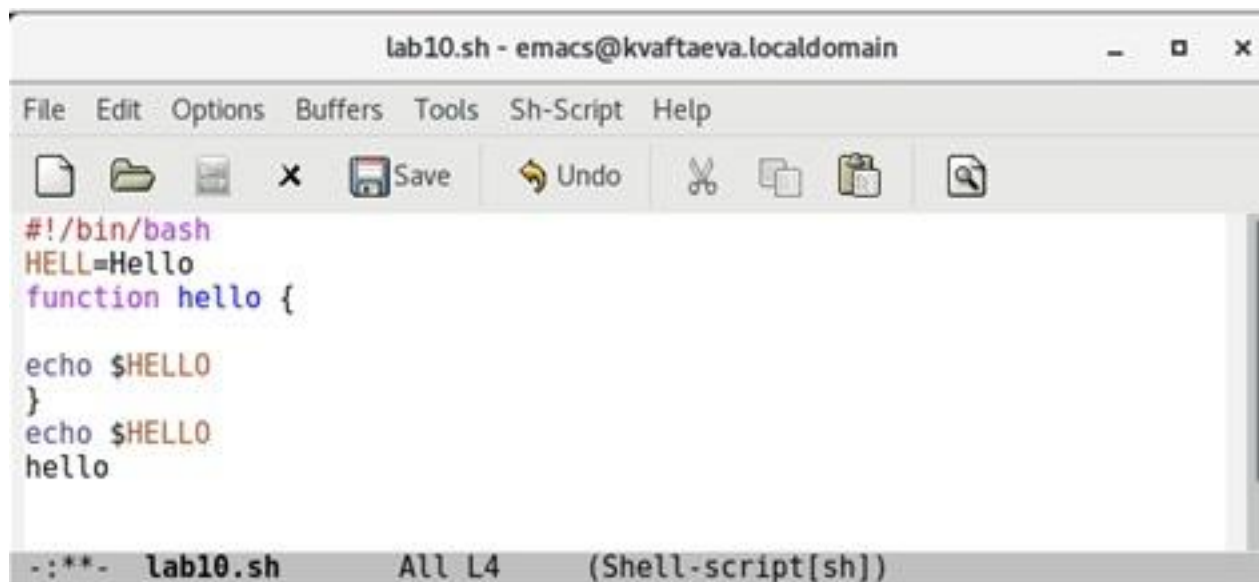


Рис.8 Сохранение файла

Видим, что файл сохранился, по появившейся внизу надписи "записан "

7. Вырезала строку "LOCAL HELLO=World" с помощью команды C-k, предварительно установив курсор на начало этой строки с помощью перемещения клавишами стрелок (Рис.9)

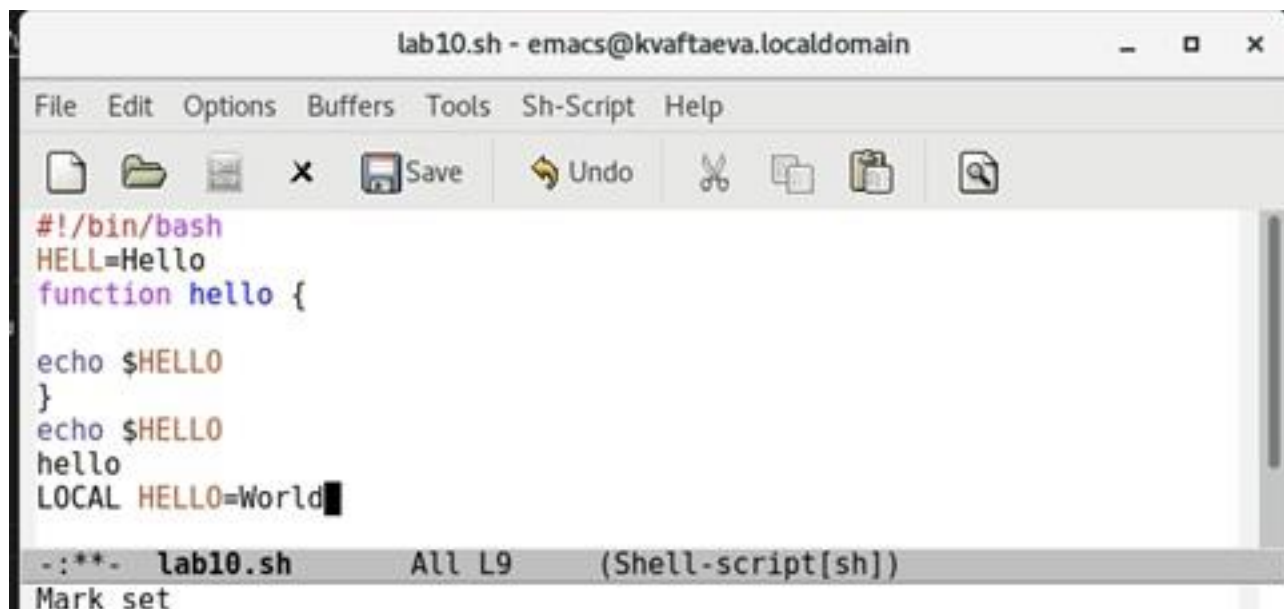


```
#!/bin/bash
HELL=Hello
function hello {
    echo $HELLO
}
echo $HELLO
hello
```

Рис.9 Удаление строки

Видим, что строки, которую мы вырезали, теперь нет

8. Переместила курсор в конец буфера (файла) с помощью комбинации клавиш Esc->. После чего вставила вырезанную ранее строку в конец файла с помощью комбинации клавиш C-y (Рис.10)



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

Рис.10 Вставка вырезанного фрагмента

Видим, что вырезанный ранее фрагмент был вставлен в конец файла

9. Выделила область текста (строку "HELL=Hello") при помощи команды C - space. Для этого сначала устанавливаем курсор на начало нужного участка и идем до конца нужной нам области клавишами стрелок (Рис.11)

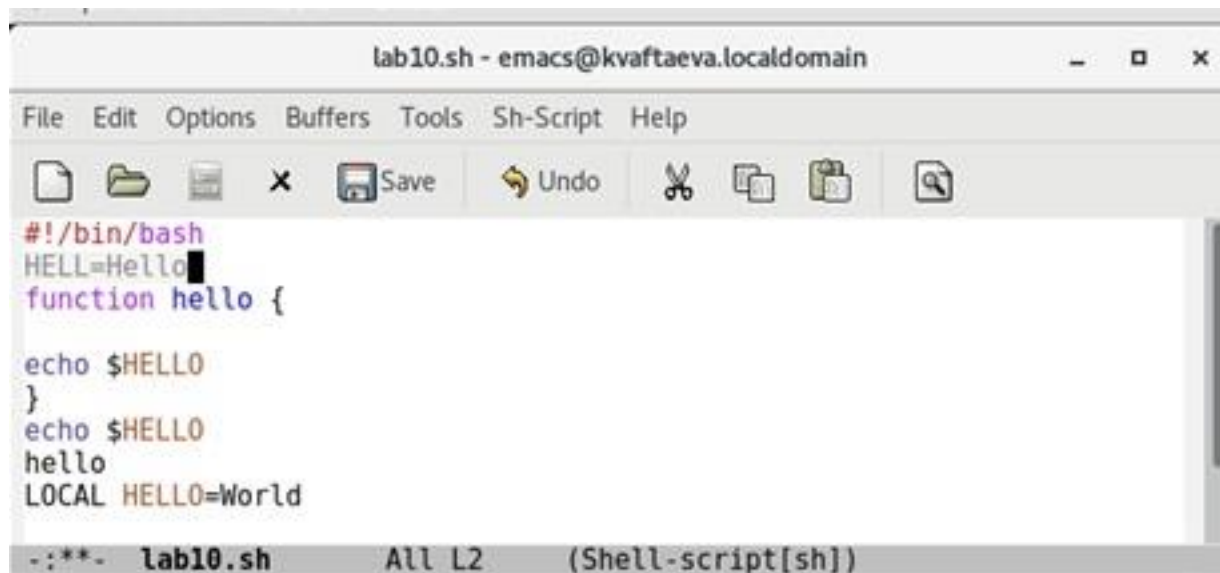


Рис.11 Выделение области

Видим, что нужный текст выделяется серым

10. Скопировала выделенную область текста в буфер обмена командой `Alt - w`. Переместила курсор в конец буфера (файла) с помощью комбинации клавиш `Esc - >`. Вставила область в конец файла командой `C - y` (Рис.12)

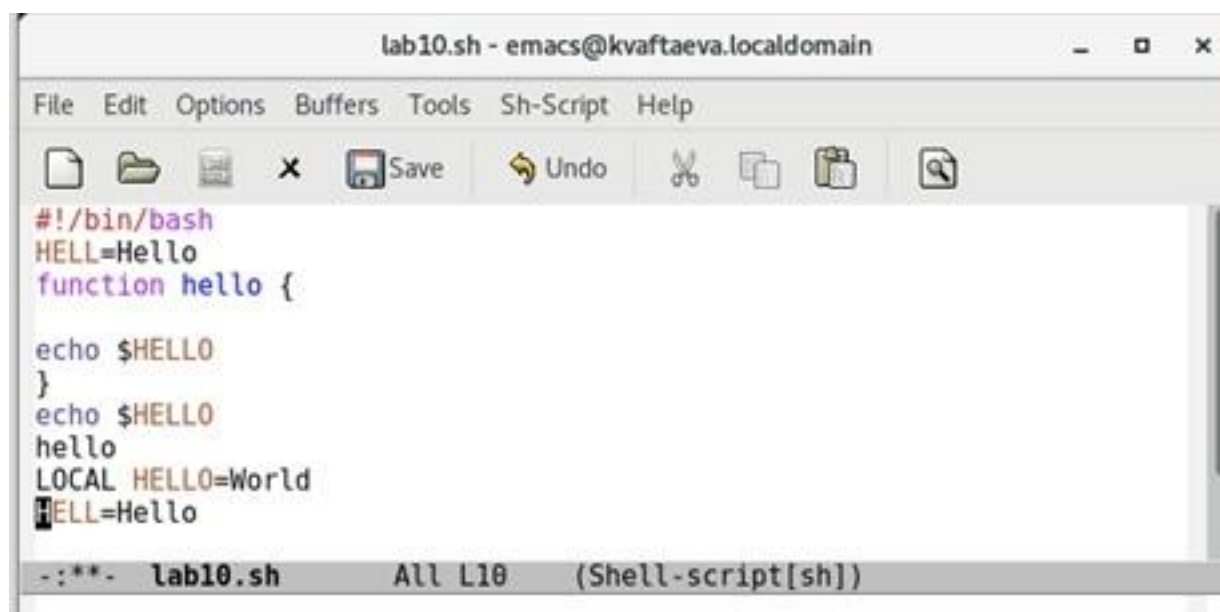


Рис.12 Копирование и вставка выделенной области

11. Вновь выделяю ту же область, что и в последних пунктах, теми же командами. Но теперь вырезаю ее комбинацией клавиш `C - w` (Рис.13)

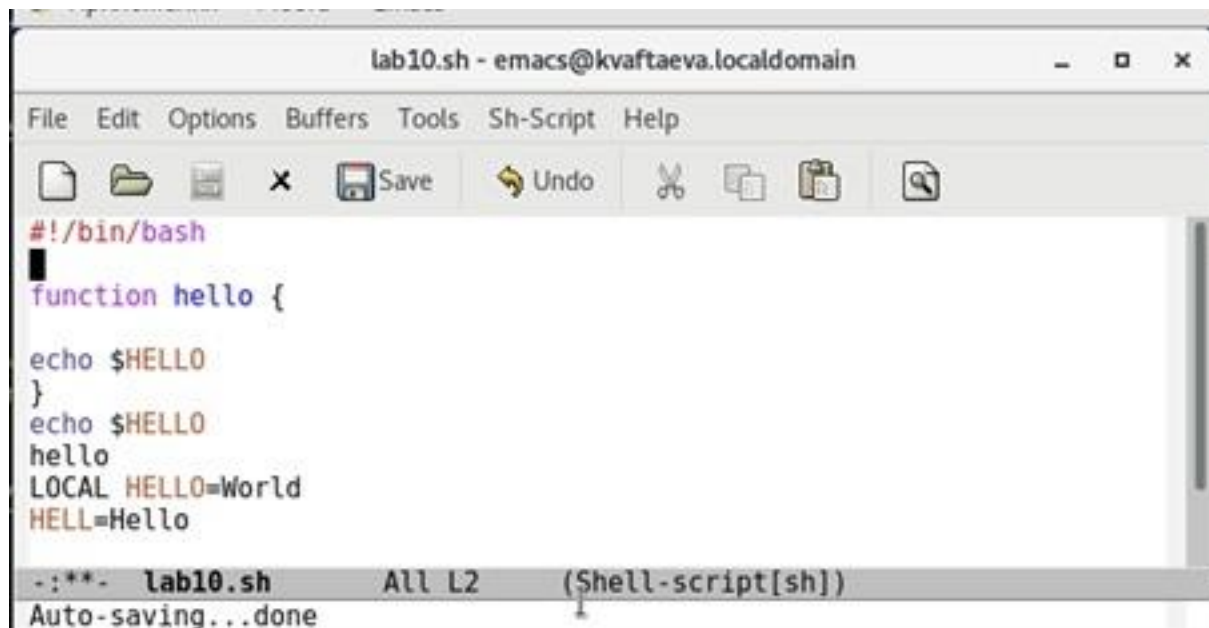


Рис.13 Вырезание выделенной области

Видим, что нужный фрагмент действительно был вырезан

12. Отменяю последнее действие комбинацией клавиш C-/ (Рис.14)

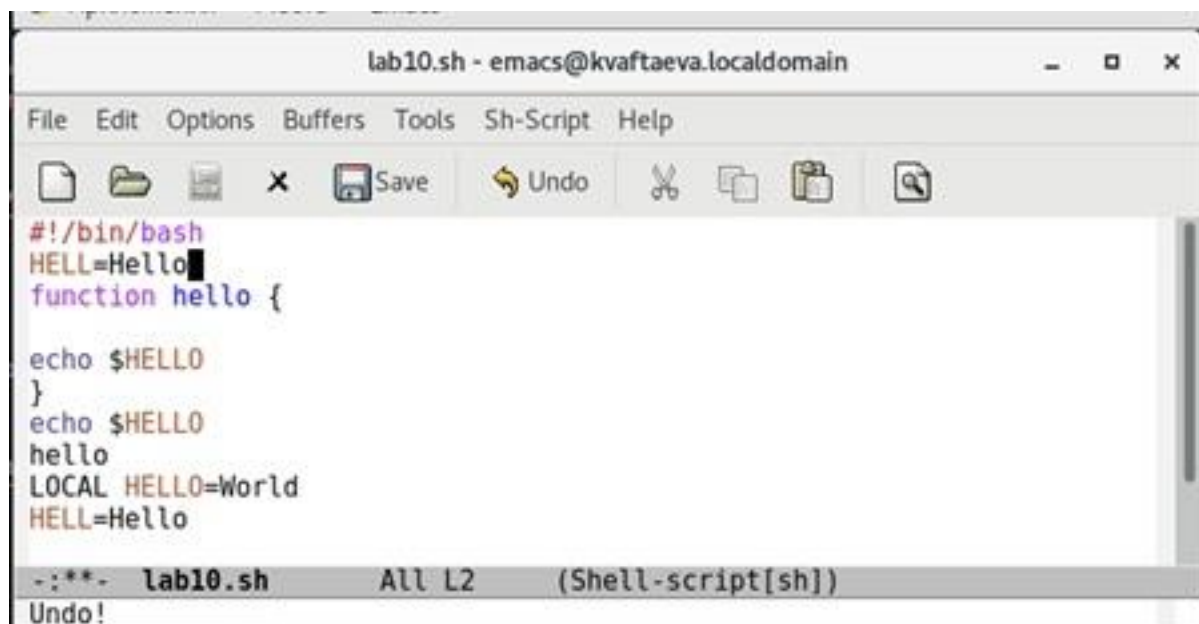


Рис.14 Отмена последнего действия

Так как последним действием было вырезание строки, после отмены эта строка снова появилась на своем месте

13. Изначальное расположение курсора перед выполнением этого пункта осталось как на Рис.14 (в следующих пунктах по перемещению курсора изначальное

расположение курсора для каждого пункта будет совпадать с его расположением после выполнения предыдущего пункта). С помощью комбинации клавиш C-a перемещаю курсор в начало строки (Рис.15)

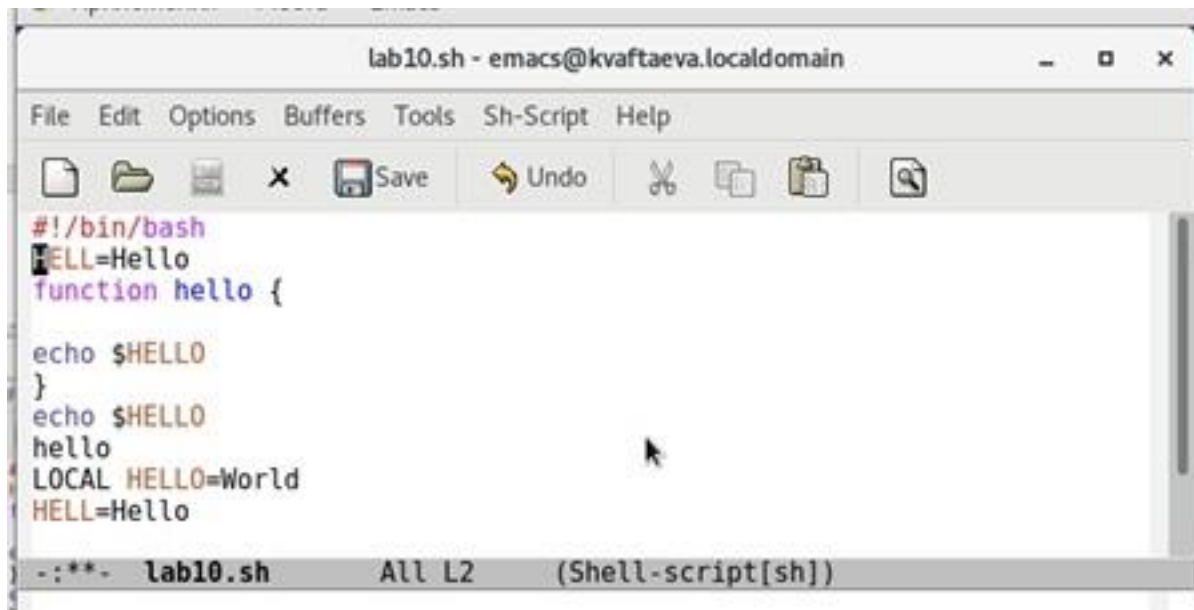


Рис.15 Перемещение курсора в начало строки

14. С помощью комбинации клавиш C-e перемещаю курсор в конец строки (Рис.16)

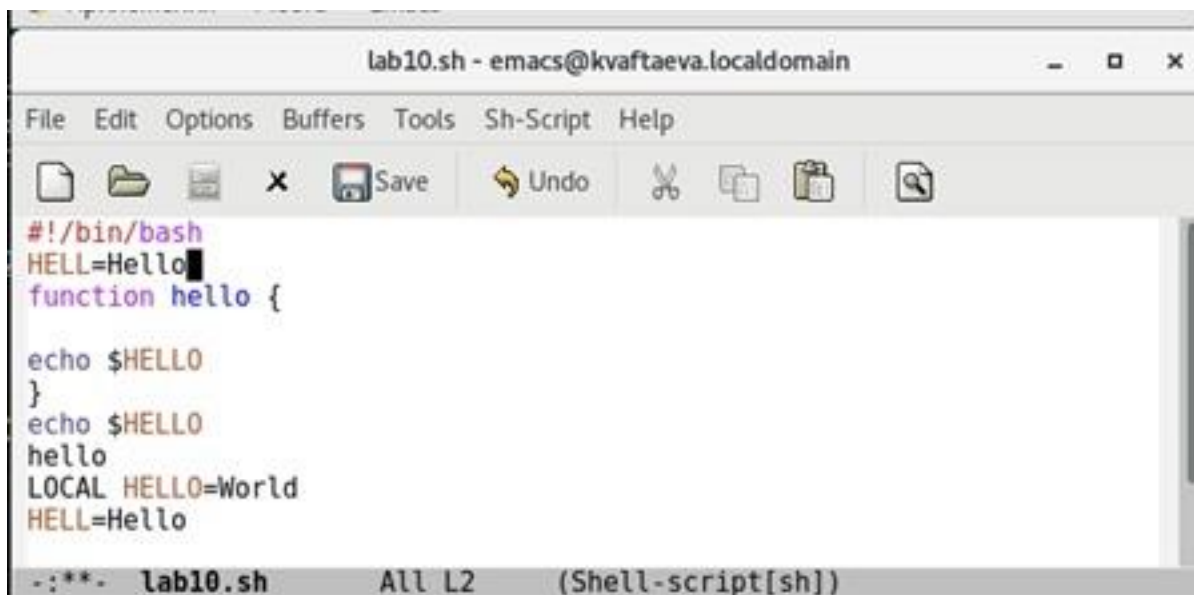


Рис.16 Перемещение курсора в конец строки

15. Переместим курсор в начало буфера (файла) комбинацией клавиш Esc - < (Рис.17)

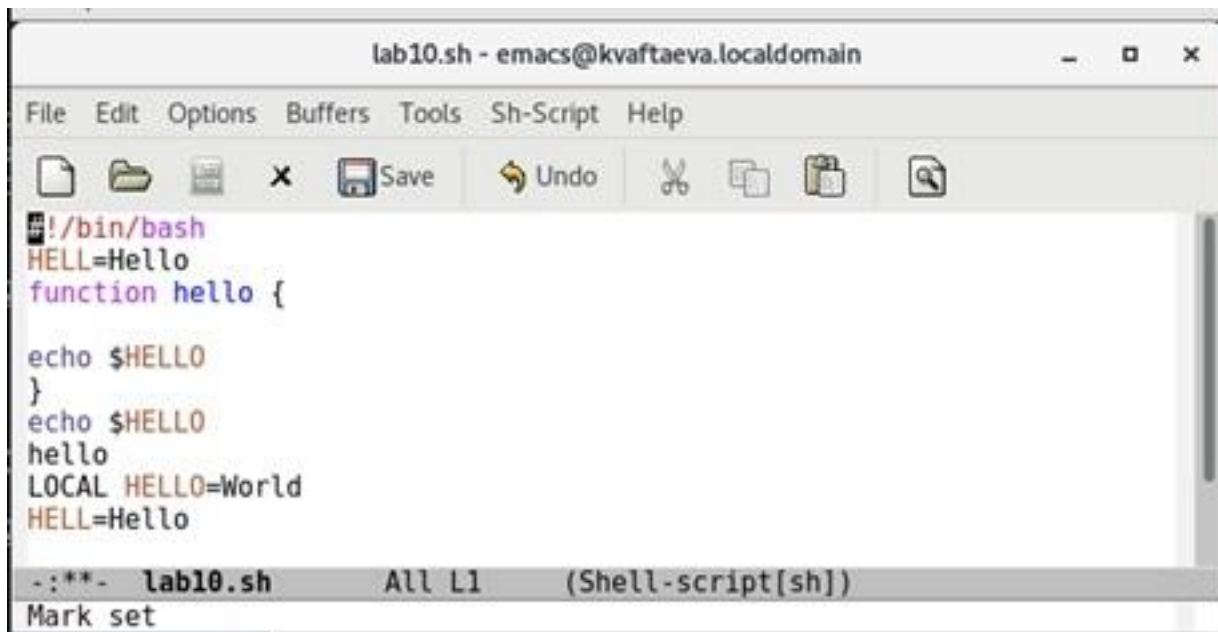


Рис.17 Перемещение курсора в начало файла

16. Переместим курсор в конец буфера (файла) комбинацией клавиш Esc - > (Рис.18)

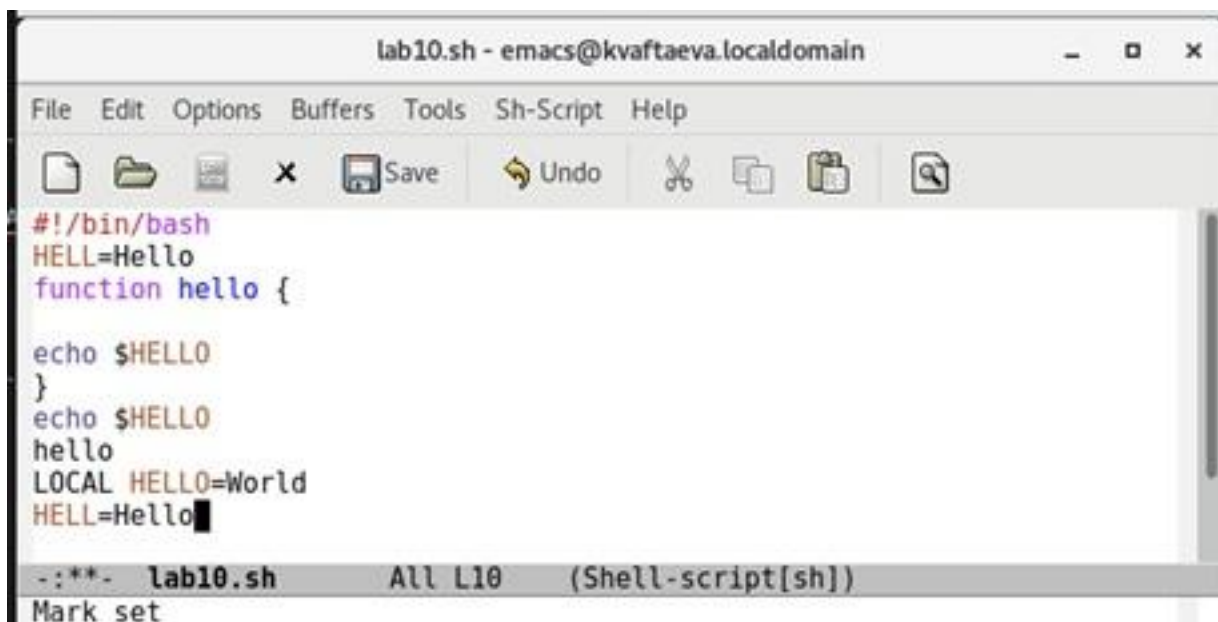


Рис.18 Перемещение курсора в конец файла

17. Вывела список активных буферов на экран с помощью комбинации клавиш C-x C-b (Рис.19)

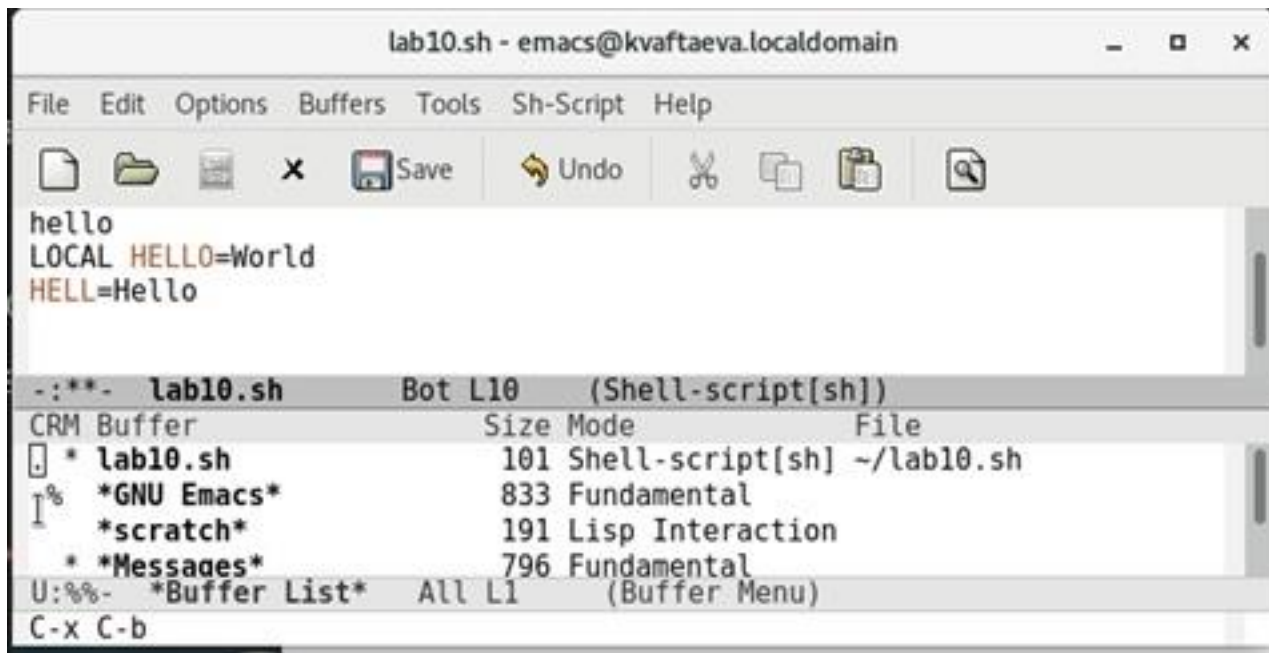


Рис.19 Вывод списка активных буферов

Видим, что внизу экрана появился список с активными буферами

18. Переместилась во вновь открытое окно со списком открытых буферов командой C-x O (Рис.20).

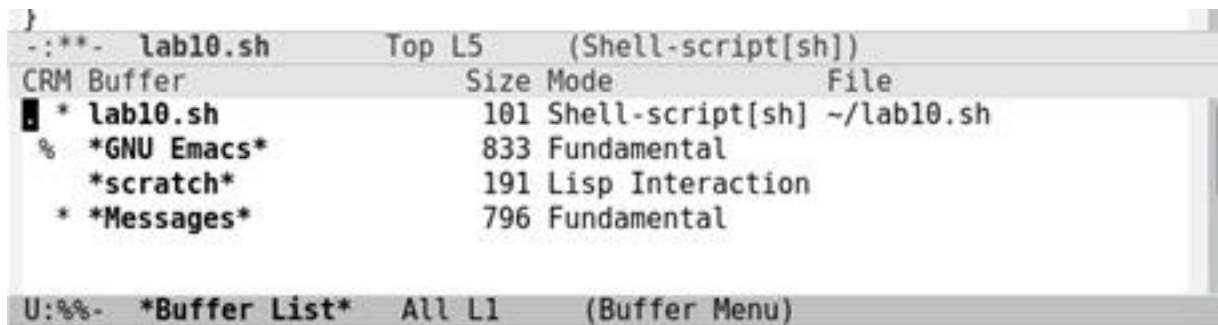


Рис.20 Перемещение в окно со списком буферов

Видим, что перешли в него по расположению курсора.

Переключился на другой буфер командой C-x b. Высвечивается фрейм, куда мы вводим название буфера, на который мы хотим переключиться. Я переключилась в буфер *scratch* (Рис.21)

```
I -:**- lab10.sh      Top L5      (Shell-script[sh])
CRM Buffer          Size Mode      File
| * lab10.sh        101 Shell-script[sh] ~/lab10.sh
% *GNU Emacs*      833 Fundamental
| *scratch*         191 Lisp Interaction
| *Messages*        796 Fundamental

U:%%- *Buffer List*  All L1      (Buffer Menu)
Switch to buffer (default *GNU Emacs*): scratch
```

Рис.21 Переключение на другой буфер

Видим, что мы действительно переместились в этот буфер (Рис.22)

```
scratch - emacs@kvaftaeva.localdomain
File Edit Options Buffers Tools Help
[Icons: New, Open, Save, Undo, Cut, Copy, Paste, Find]
#!/bin/bash
HELL=Hello
function hello {
  echo $HELLO
}

I -:**- lab10.sh      Top L5      (Shell-script[sh])

U:--- scratch        All L1      (Fundamental)
```

Рис.22 Буфер scratch

19. Закрывает окно с буфером **lab10.sh** с помощью комбинации клавиш C-x 0. Видим, что на экране соотслся только буфер **scratch** (Рис.23)

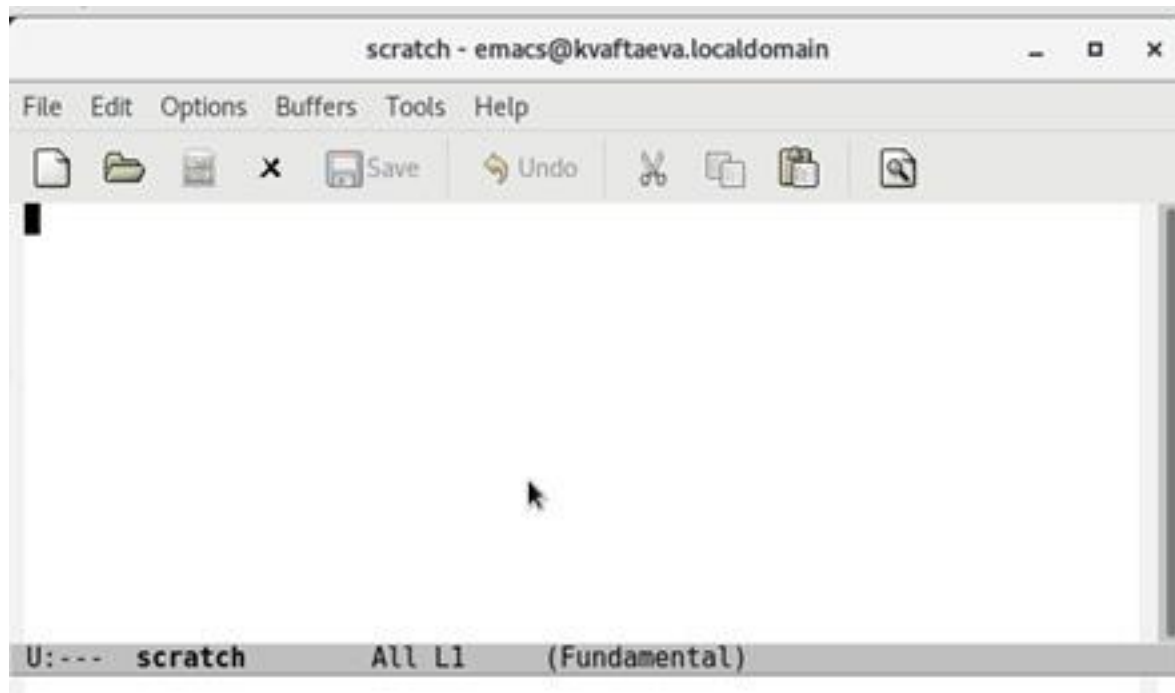


Рис.23 Заккрытие окна

20. Теперь можем переключиться между буферами без вывода их списка на экран с помощью команды `C-x b`, вводя в появившийся фрейм название нужного буфера. Я переключаюсь на буфер **lab10.sh** (Рис.24)

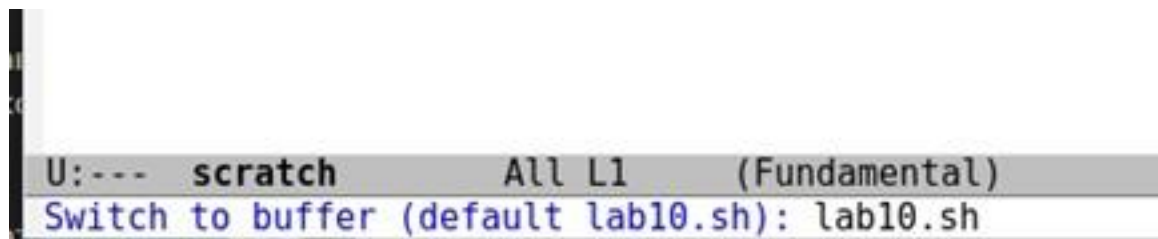


Рис.24 Переключение между буферами без вывода списка

Видим, что действительно переместились в нужный буфер (Рис.25)

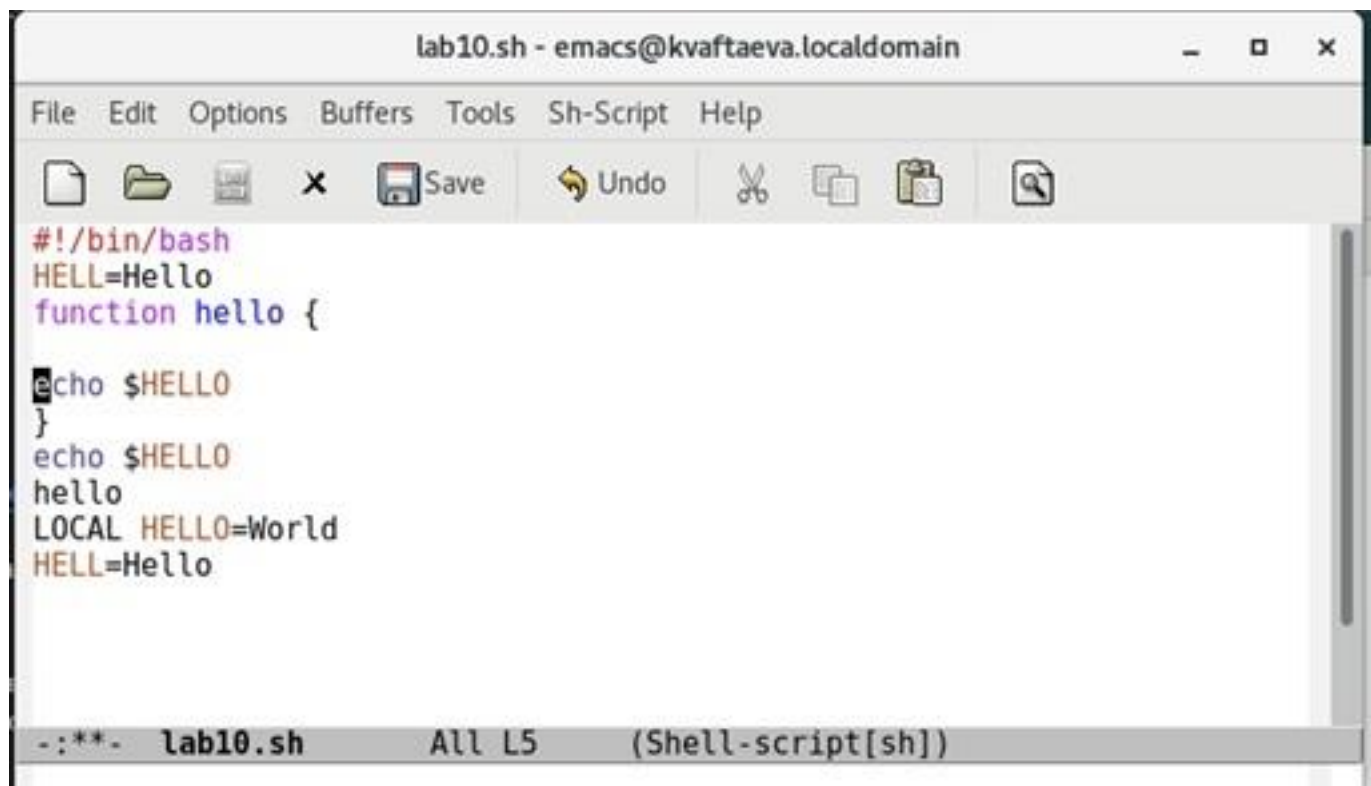


Рис.25 Буфер lab10.sh

21. Поделила фрейм на 4 части: разделила фрейм на два окна по вертикали комбинацией клавиш C-x 3, а затем каждое из этих окон на две части по горизонтали комбинацией клавиш C-x 2 (Рис.26)

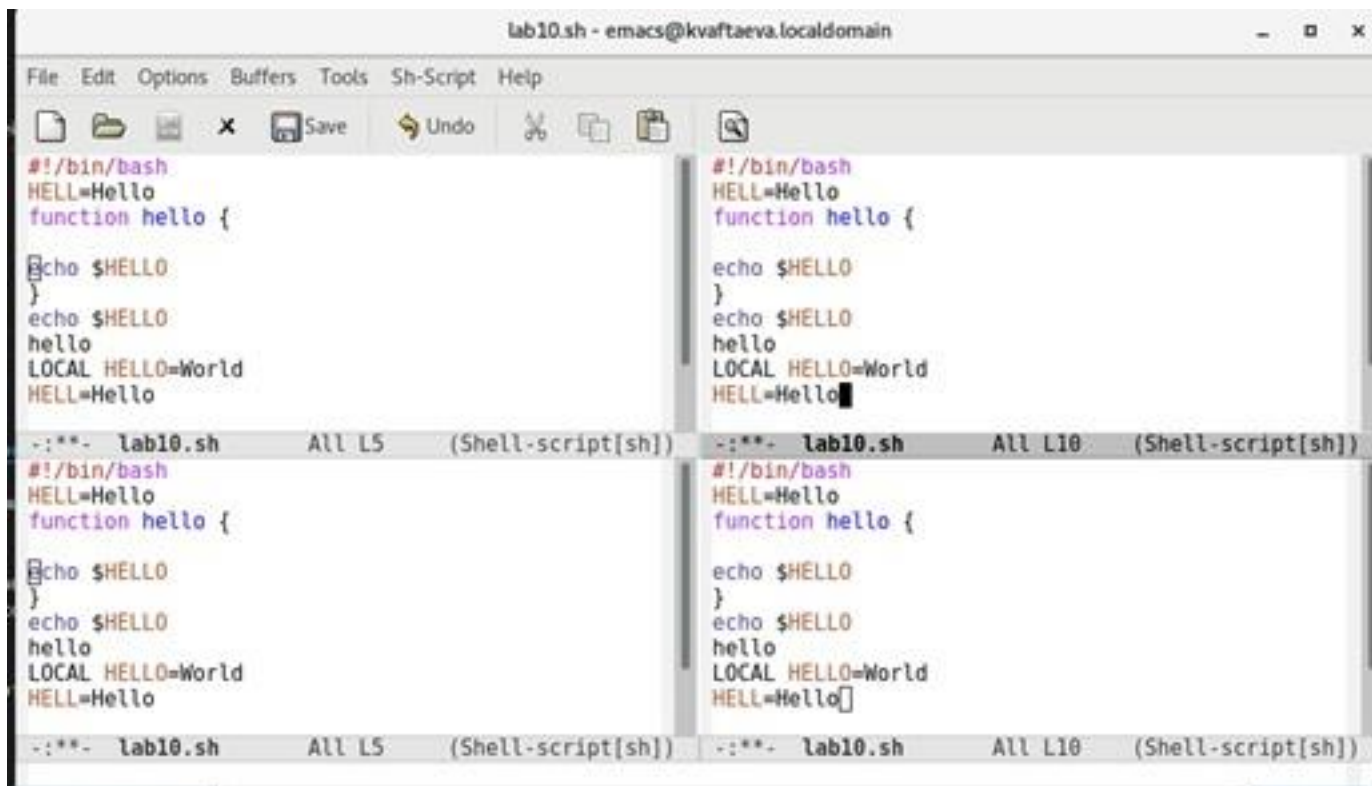


Рис.26 Разделение фрейма

Видим, что действительно произошло разделение на 4 части

22. В каждом из четырёх созданных окон откроем новый буфер (файл) newfile.txt командой C-x C-f, как в начале лабораторной работы (Рис.27)

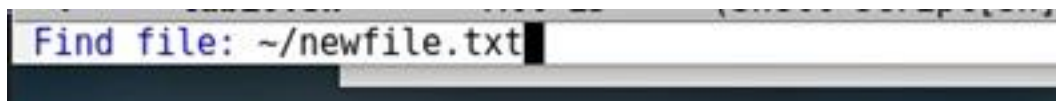


Рис.27 Создаем новый файл

Видим, что этот файл теперь создан в каждом из четырех окон (Рис.28)

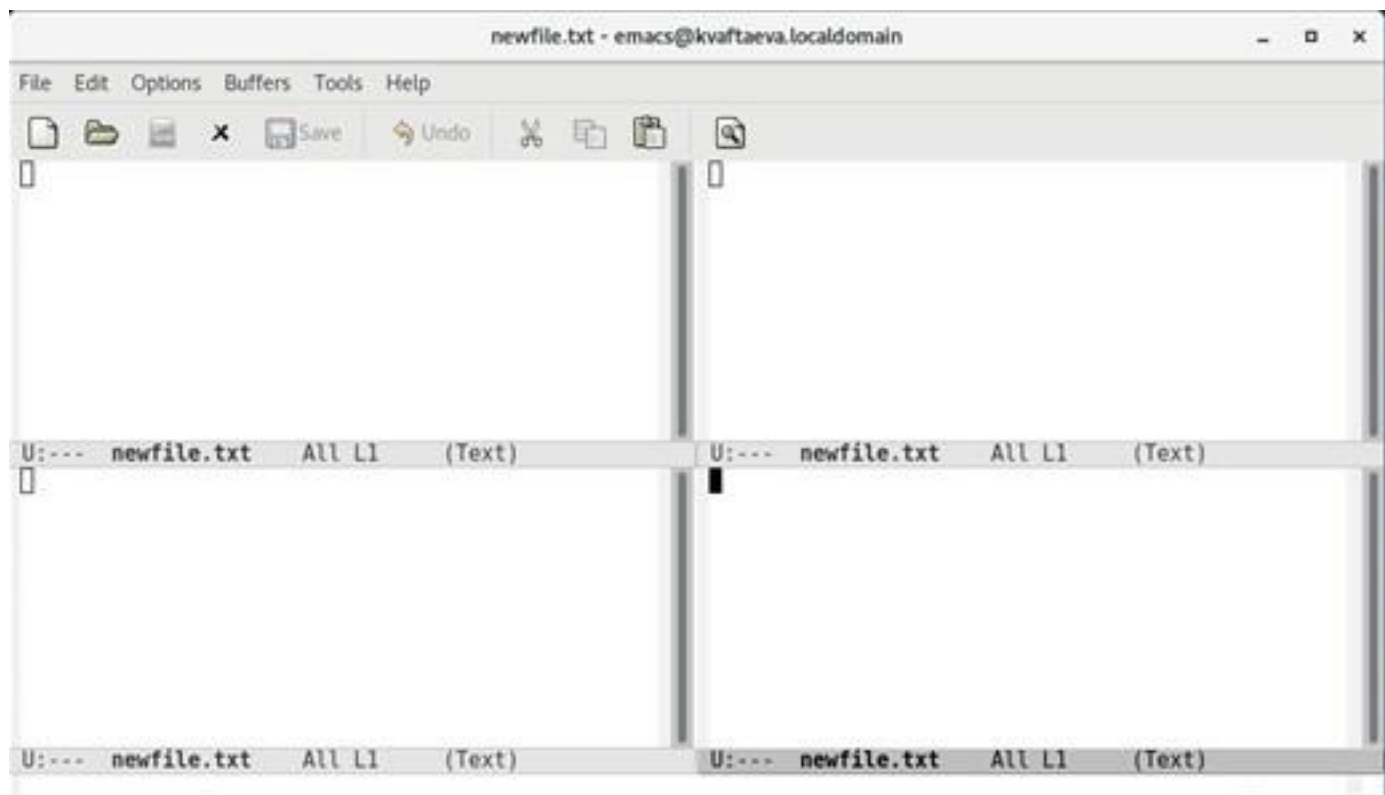


Рис.28 Создаем новый файл в четырех окнах

Теперь записываем несколько строк текста в одном из окон, текст автоматически появится в каждом окне (Рис.29)



Рис.29 Запись текста в новый файл

23. Закрываю три окна комбинацией клавиш C-x 0, оставляя одно для дальнейшей работы (Рис.30)

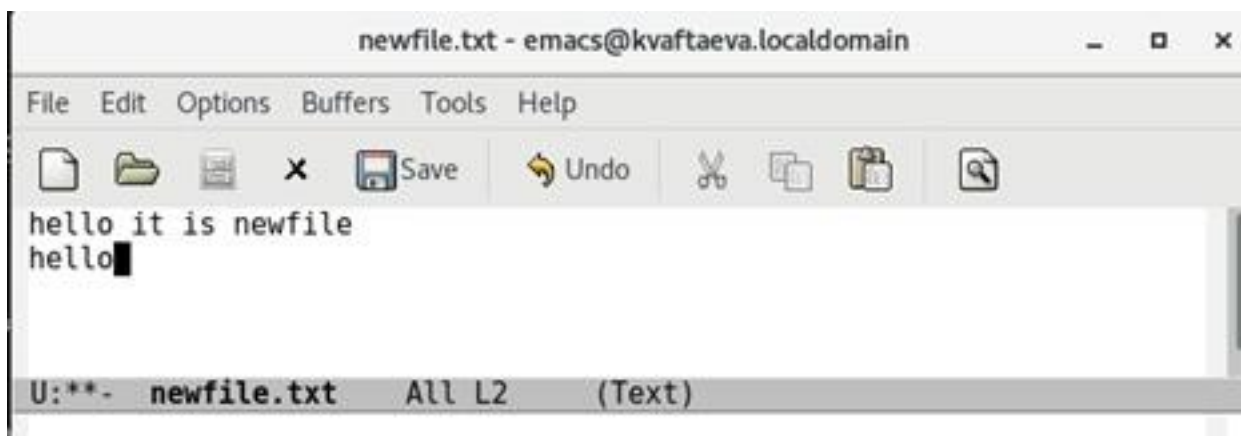


Рис.30 Окно для дальнейшей работы

24. Переключаюсь в режим поиска комбинацией клавиш C-s. У нас появляется строка, куда я ввожу слово, которое ищу (Рис.31)

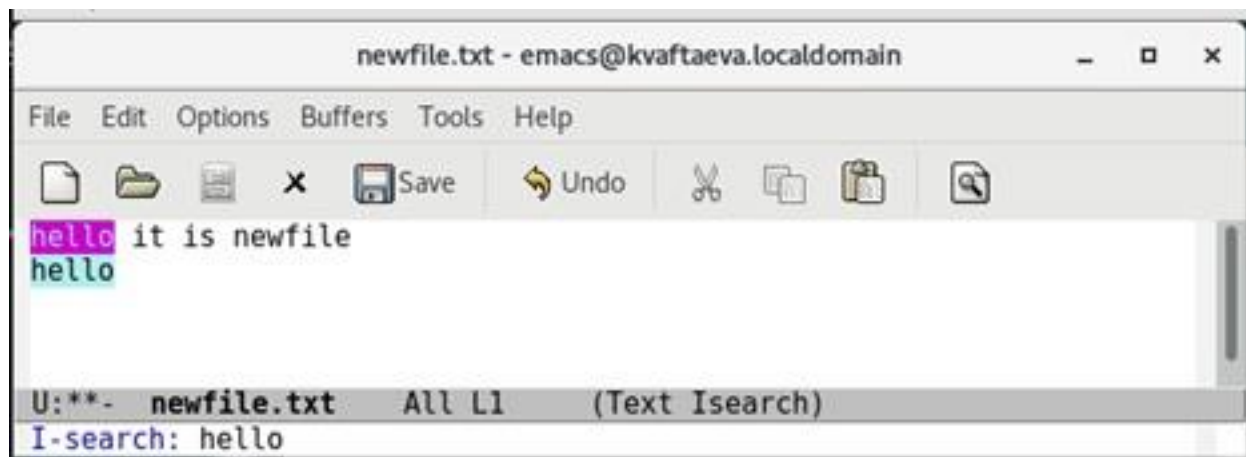


Рис.31 Режим поиска

Видим, что найденные слова подсвечиваются

25. С помощью той же комбинации клавиш (C-s) переключаемся между результатами поиска (Рис.32)

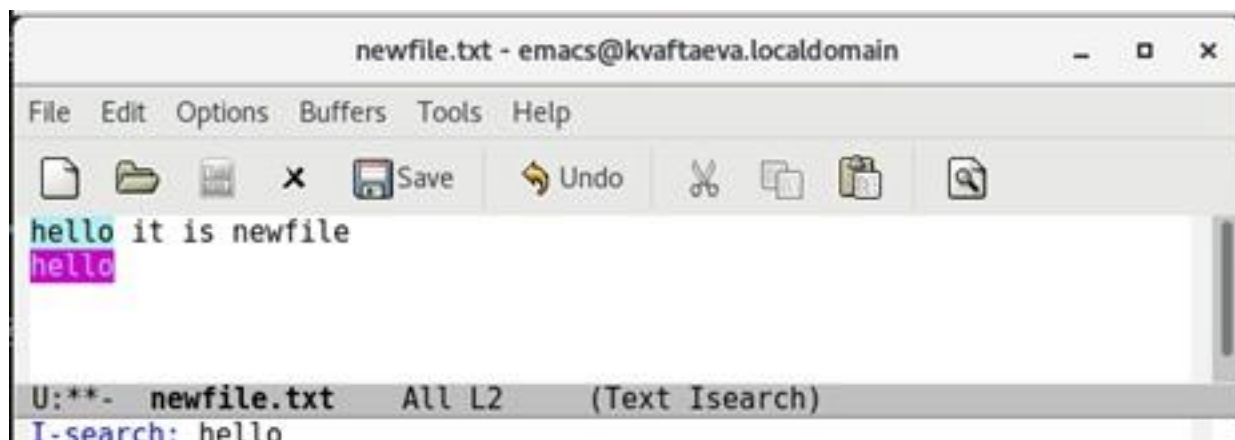


Рис.32 Переключение между результатами

Видим, что переключились на второе найденное слово, так как он теперь выделено фиолетовым

26. Вышла из режима поиска комбинацией клавиш C-g
27. Перешла в режим поиска и замены с помощью комбинации клавиш Esc-%, в появившуюся строку запроса ввела текст, который хочу найти и заменить (Рис.33) и нажала Enter.

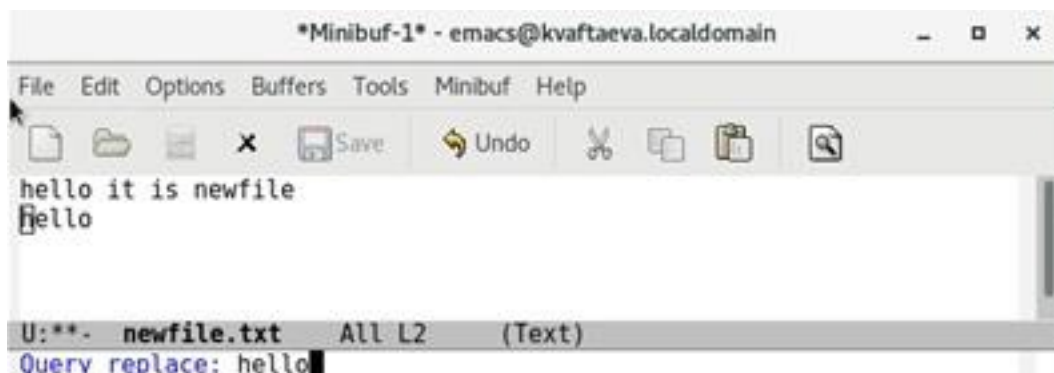


Рис.33 Запрос на замену

Затем ввела текст для замены (Рис.34)

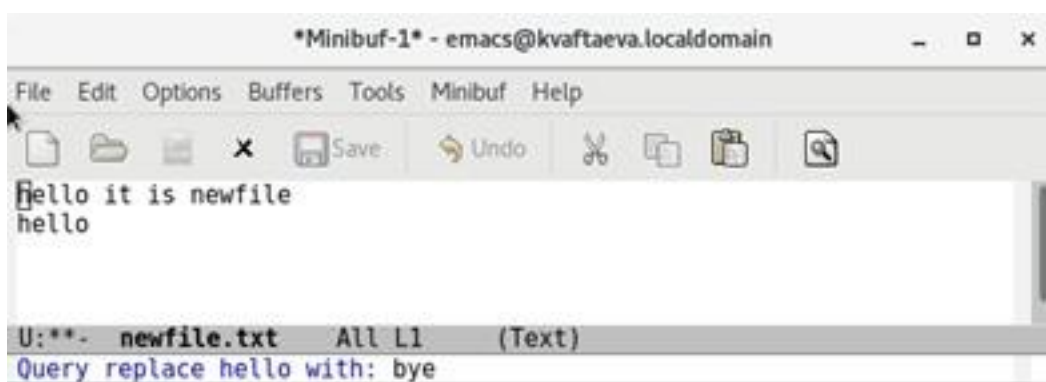


Рис.34 Текст для замены

После того как были подсвечены результаты поиска и выделено слово для замены (Рис.35), нажала на ! для подтверждения замены

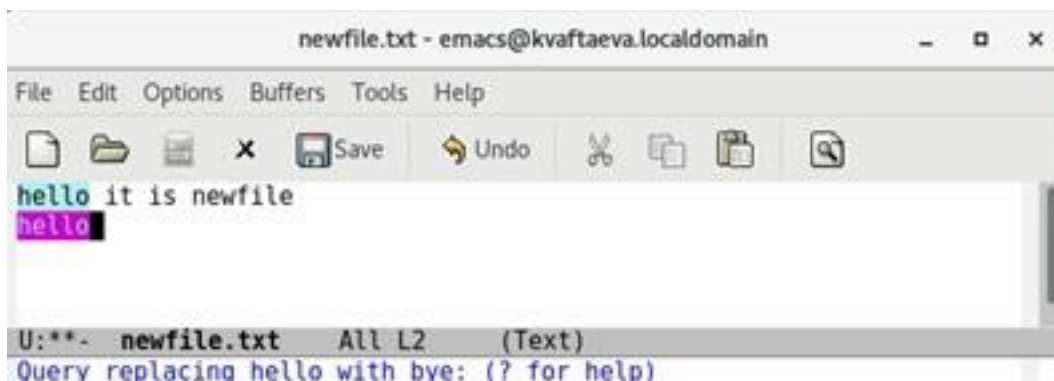


Рис.35 Подсвеченные результаты поиска

Видим, что замена прошла успешно (Рис.36)

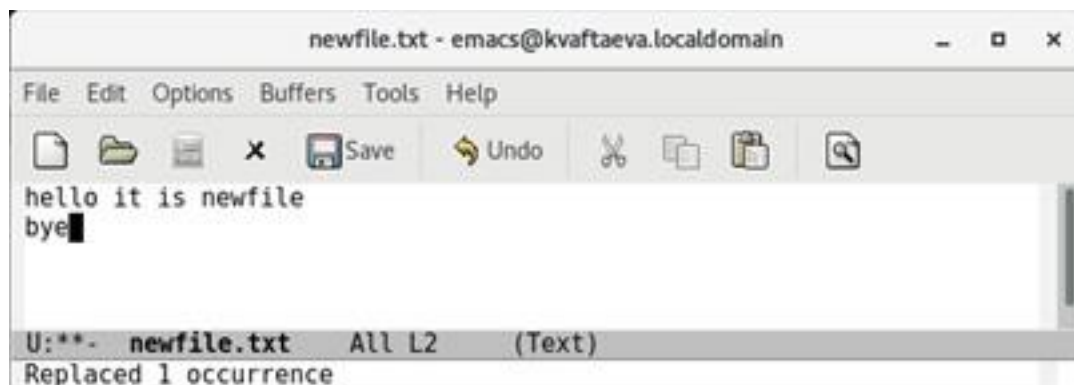


Рис.36 Результат замены

28. Испробовала другой режим поиска, нажав M-s и введя для поиска слово "hello" (Рис.37)

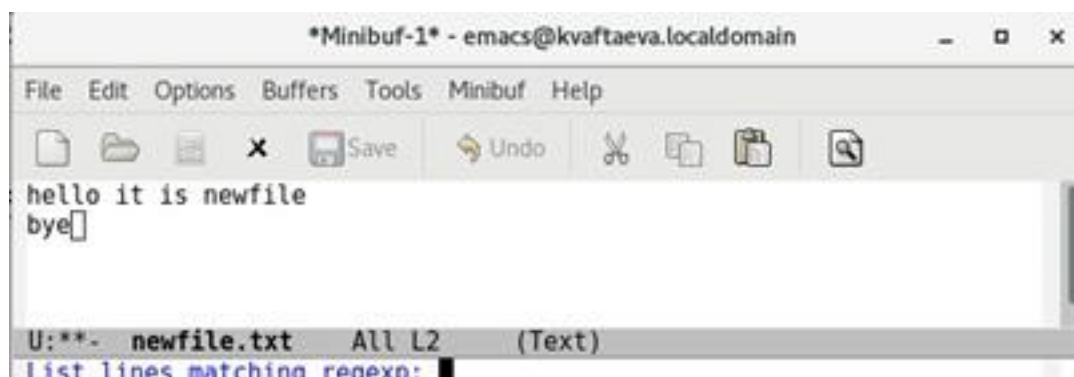


Рис.37 Новый режим поиска, запрос

Видим, что этот режим отличается тем, что ищет не слово, а строку, где находится это слово (Рис.38)

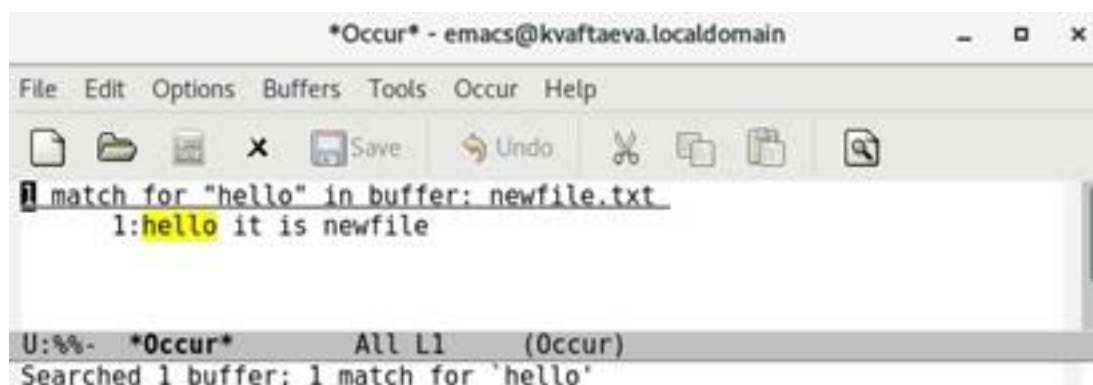


Рис.38 Новый режим поиска, результат

Контрольные вопросы: [1]

1. Emacs представляет собой мощный экраный редактор текста, написанный на языке высокого уровня Elisp
2. Сложности могут возникнуть, так как на клавиатуре для IBM PC совместимых ПК клавиши `Ctrl` нет, но вместо нее можно использовать `Alt`. Также сложно запомнить, какое сочетание клавиш для чего предназначено.
3. Буфер - это область где мы набираем текст. Окно - область, которая объединяет открытые буферы.
4. Да
 - `% GNU Emacs 844 Fundamental`
 - `scratch 191 Lisp Interaction`
 - `%* Messages 5257 Messages`
 - `% Quail Completions 0 Fundamental`
5. `Ctrl,C,Shift,,]` и `,Ctrl,C Ctrl,Shift,,]`
6. По вертикали C-x 3, по горизонтали C-x 2
7. В файле Emacs
8. Кнопка `backspace` = C-k и ее можно переназначить.
9. Emacs, потому что в нём больше функций и он больше похож на те редакторы, в которых я привыкла работать.

Заключение:

Таким образом, все пункты задания были выполнены верно. Все возникшие по мере выполнения проблемы и ошибки были решены. Ознакомление с новым редактором прошло успешно.

Вывод:

Я познакомилась с операционной системой **Linux**. Получила практические навыки работы с редактором **Emacs**.

Библиографический список:

[1]: Редактор emacs

[2]: Описание лабораторной работы №10