

Лабораторная работа № 10.

Текстовый редактор vi

Дагделен Зейнап Реджеповна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	14
6	Ответы на контрольные вопросы	15

Список иллюстраций

4.1	Создание файла через vi	8
4.2	Запись в файл	8
4.3	Режим последней строки	9
4.4	Сохранение и выход	9
4.5	Обновление прав	9
4.6	Открытие с помощью текстового редактора	9
4.7	Редактирование файла	10
4.8	Редактирование файла	10
4.9	Редактирование файла	11
4.10	Редактирование файла	11
4.11	Отмена последнего действия	12
4.12	Сохранение и выход	13
6.1	Граф взаимосвязи режимов работы редактора vi	18

Список таблиц

1 Цель работы

Цель данной лабораторной работы - познакомиться с операционной системой Linux, получить практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.

2 Задание

1. Ознакомиться с теоретическим материалом.
2. Ознакомиться с редактором vi.
3. Выполнить упражнения, используя команды vi.

3 Теоретическое введение

В большинстве дистрибутивов Linux в качестве текстового редактора по умолчанию устанавливается интерактивный экранный редактор vi (Visual display editor). Редактор vi имеет три режима работы:

- командный режим — предназначен для ввода команд редактирования и навигации по редактируемому файлу;
 - режим вставки — предназначен для ввода содержания редактируемого файла;
 - режим последней (или командной) строки — используется для записи изменений в файл и выхода из редактора.
- Для вызова редактора vi необходимо указать команду vi и имя редактируемого файла: vi При этом в случае отсутствия файла с указанным именем будет создан такой файл.
- Переход в командный режим осуществляется нажатием клавиши Esc.
 - Для выхода из редактора vi необходимо перейти в -режим последней строки: находясь в командном режиме, нажать Shift-; (по сути символ : — двоеточие), затем:
 - набрать символы wq, если перед выходом из редактора требуется записать изменения в файл;
 - набрать символ q (или q!), если требуется выйти из редактора без сохранения.

4 Выполнение лабораторной работы

Создаю директорию, в которой буду работать, с помощью команды `mkdir`, перехожу в нее с помощью команды `cd`, создаю и открываю для редактирования файл с помощью встроенного текстового редактора `vi` (рис. fig. 4.1).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ mkdir -p ~/work/os/lab06
zrdagdelen@zrdagdelen-MacBookPro:~$ cd ~/work/os
zrdagdelen@zrdagdelen-MacBookPro:~/work/os$ ls
lab06
zrdagdelen@zrdagdelen-MacBookPro:~/work/os$ cd lab06
zrdagdelen@zrdagdelen-MacBookPro:~/work/os/lab06$ vi hello.sh
```

Рис. 4.1: Создание файла через `vi`

Нажимаю `i`, чтобы начать редактирование и добавляться текст (режим добавления), далее добавляю текст (рис. fig. 4.2).

```
zrdagdelen@zrdagdelen-MacBookPro: ~/work/os/lab06
#!/bin/bash
HELL=Hello
function hello {
LOCAL HELLO=World
echo $HELLO
}
echo $HELLO
hello
```

Рис. 4.2: Запись в файл

Нажимаю `esc`, чтобы выйти из режима добавления и перейти в командный режим. Перехожу в режим последней строки (рис. fig. 4.3).



Рис. 4.3: Режим последней строки

Ввожу `w` для сохранения и `q` для выхода и нажимаю `enter`, после чего файл сохраняется с изменениями и я возвращаюсь в терминал (рис. fig. 4.4).

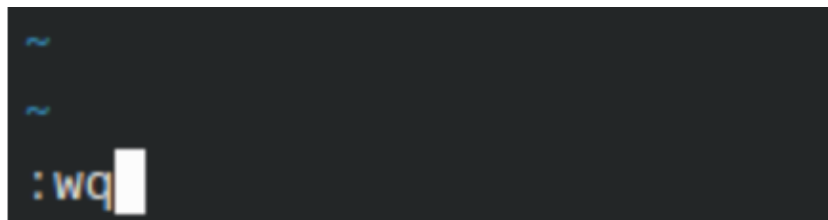


Рис. 4.4: Сохранение и выход

С помощью `chmod` добавляю права на исполнение файла, делая его исполняемым (рис. fig. 4.5).

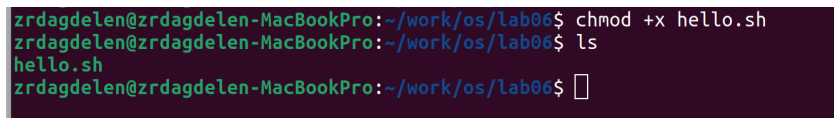


Рис. 4.5: Обновление прав

Снова открываю этот файл с помощью текстового редактора `vi` (рис. fig. 4.6).

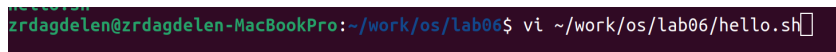


Рис. 4.6: Открытие с помощью текстового редактора

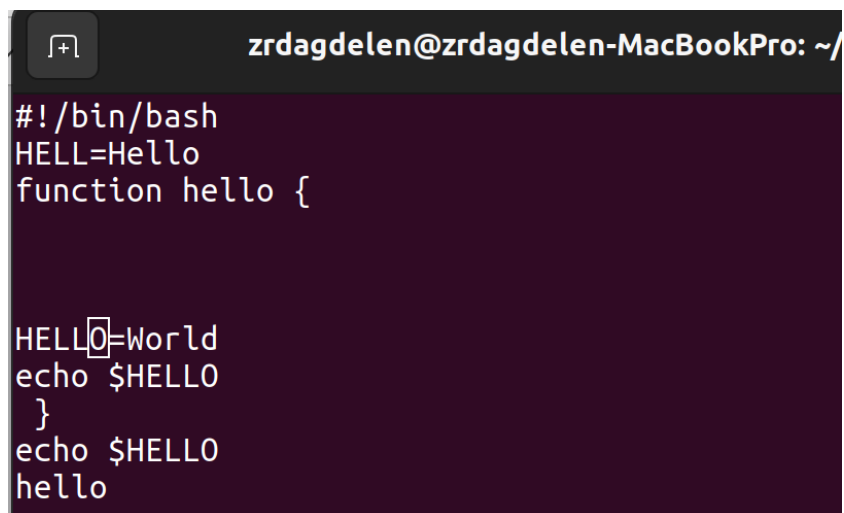
Изменяю местоположение курсора (рис. fig. 4.7).



```
zrdagdelen@zrdagdelen-MacBookPro: ~  
#!/bin/bash  
HELLO=Hello  
function hello {  
  LOCAL HELLO=World  
  echo $HELLO  
}  
echo $HELLO  
hello  
~
```

Рис. 4.7: Редактирование файла


В режиме добавления (клавиша i) удаляю слово LOCAL, снова перехожу в командный режим с помощью esc (рис. fig. 4.8).



```
zrdagdelen@zrdagdelen-MacBookPro: ~/...  
#!/bin/bash  
HELLO=Hello  
function hello {  
  
  HELLO=World  
  echo $HELLO  
}  
echo $HELLO  
hello
```

Рис. 4.8: Редактирование файла

В режиме добавления (клавиша i) добавляю слово local, снова перехожу в командный режим с помощью esc (рис. fig. 4.9).

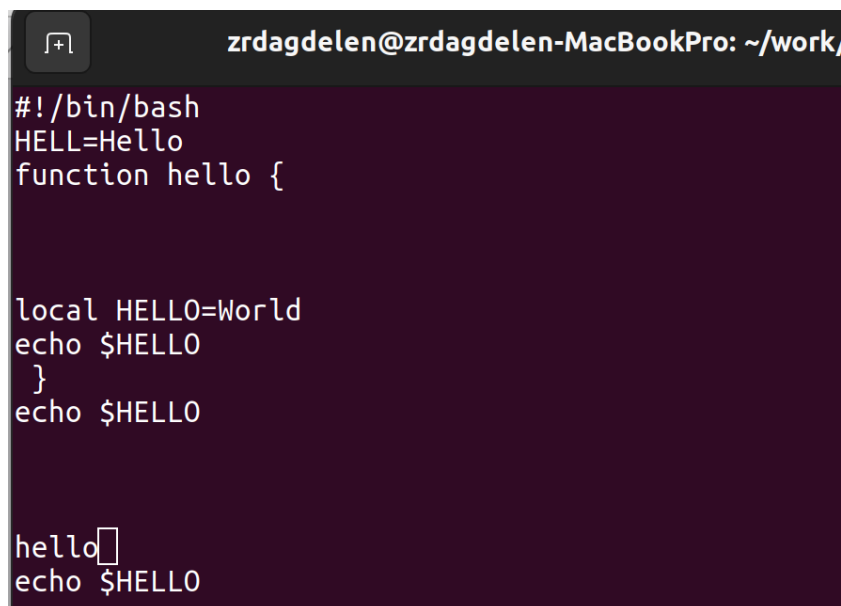


```
zrdagdelen@zrdagdelen-MacBookPro: ~/
#!/bin/bash
HELL=Hello
function hello {

local HELLO=World
echo $HELL0
}
echo $HELL0
hello
```

Рис. 4.9: Редактирование файла

В режиме добавления (клавиша i) перехожу в конец последней строки и добавляю новую строку, снова перехожу в командный режим с помощью esc. (рис. fig. 4.10). После этого я вернулась в режим добавления, удалила строку и вернулась в командный режим.



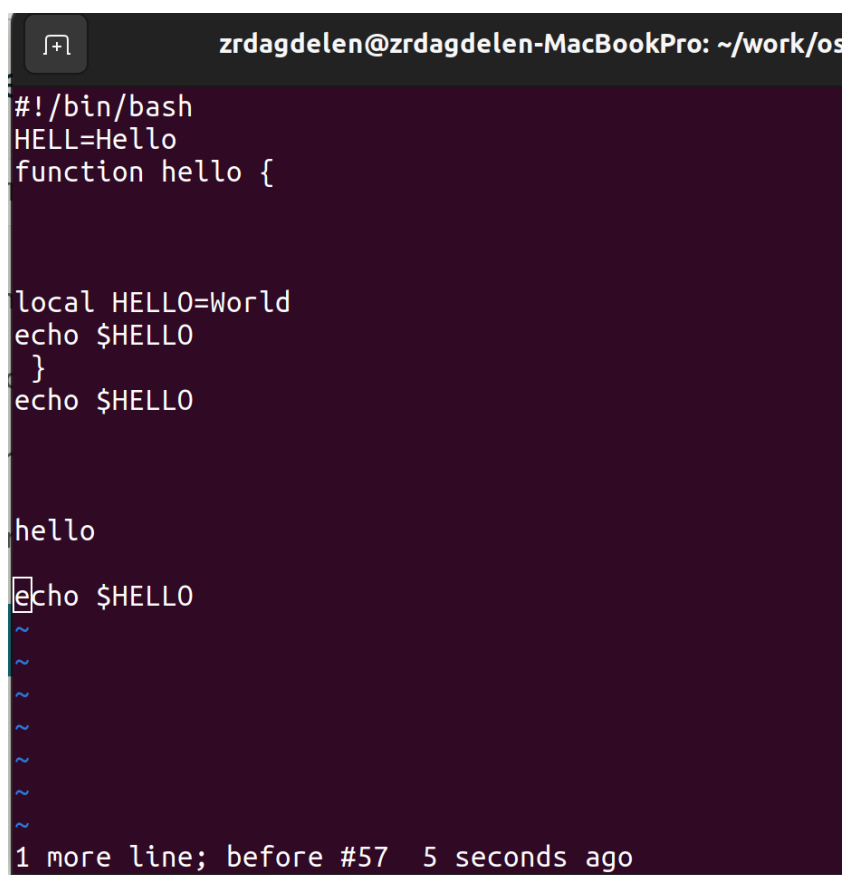
```
zrdagdelen@zrdagdelen-MacBookPro: ~/work,
#!/bin/bash
HELL=Hello
function hello {

local HELLO=World
echo $HELL0
}
echo $HELL0

hello
echo $HELL0
```

Рис. 4.10: Редактирование файла

В командном режиме я нажала “u” и отменила последнее действие, удаление строки (рис. fig. 4.11).



```
zrdagdelen@zrdagdelen-MacBookPro: ~/work/os
#!/bin/bash
HELL=Hello
function hello {

local HELLO=World
echo $HELL
}
echo $HELL

hello
echo $HELL
~
~
~
~
~
~
~
1 more line; before #57 5 seconds ago
```

Рис. 4.11: Отмена последнего действия

Перешла в режим последней строки и ввела w для сохранения файла и q для выхода, нажала enter и файл закрылся (рис. fig. 4.12).



A terminal window titled "zrdagdelen@zrdagdelen-MacBookPro: ~/work/o" with a window control icon. The terminal displays a bash script and its execution. The script defines a function "hello" that sets a local variable "HELLO" to "World", echoes "\$HELLO", and then echoes "\$HELLO" again. The script is then executed by typing "hello" and "echo \$HELLO". The output shows "World" and "World" respectively. The prompt is "~" and the cursor is at the end of the line ":wq".

```
#!/bin/bash
HELL=Hello
function hello {

local HELLO=World
echo $HELLO
}
echo $HELLO

hello

echo $HELLO
~
~
~
~
~
~
~
:wq
```

Рис. 4.12: Сохранение и выход

5 Выводы

При выполнении данной лабораторной работы я познакомилась с операционной системой Linux, получила практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.

6 Ответы на контрольные вопросы

1. Дайте краткую характеристику режимам работы редактора vi.

- командный режим — предназначен для ввода команд редактирования и навигации по редактируемому файлу;
- режим вставки — предназначен для ввода содержания редактируемого файла;
- режим последней (или командной) строки — используется для записи изменений в файл и выхода из редактора.

2. Как выйти из редактора, не сохраняя произведённые изменения?

Можно нажимать символ q (или q!), если требуется выйти из редактора без сохранения.

3. Назовите и дайте краткую характеристику командам позиционирования.

- 0 (ноль) — переход в начало строки;
- \$ — переход в конец строки;
- G — переход в конец файла;
- n G — переход на строку с номером n.

4. Что для редактора vi является словом?

Редактор vi предполагает, что слово - это строка символов, которая может включать в себя буквы, цифры и символы подчеркивания.

5. Каким образом из любого места редактируемого файла перейти в начало (конец) файла?

С помощью G — переход в конец файла

6. Назовите и дайте краткую характеристику основным группам команд редактирования.

- Вставка текста – a — вставить текст после курсора; – A — вставить текст в конец строки; – i — вставить текст перед курсором; – n i — вставить текст n раз; – I — вставить текст в начало строки.
- Вставка строки – o — вставить строку под курсором; – O — вставить строку над курсором.
- Удаление текста – x — удалить один символ в буфер; – d w — удалить одно слово в буфер; – d \$ — удалить в буфер текст от курсора до конца строки; – d 0 — удалить в буфер текст от начала строки до позиции курсора; – d d — удалить в буфер одну строку; – n d d — удалить в буфер n строк.
- Отмена и повтор произведённых изменений – u — отменить последнее изменение; – . — повторить последнее изменение.
- Копирование текста в буфер – Y — скопировать строку в буфер; – n Y — скопировать n строк в буфер; – y w — скопировать слово в буфер.
- Вставка текста из буфера – p — вставить текст из буфера после курсора; – P — вставить текст из буфера перед курсором.
- Замена текста – c w — заменить слово; – n c w — заменить n слов; – c \$ — заменить текст от курсора до конца строки; – r — заменить слово; – R — заменить текст.
- Поиск текста – / текст — произвести поиск вперёд по тексту указанной строки символов текст; – ? текст — произвести поиск назад по тексту указанной строки символов текст.

7. Необходимо заполнить строку символами \$. Каковы ваши действия?

Перейти в режим вставки.

8. Как отменить некорректное действие, связанное с процессом редактирования?

C — отменить последнее изменение

9. Назовите и дайте характеристику основным группам команд режима последней строки.

Режим последней строки — используется для записи изменений в файл и выхода из редактора.

10. Как определить, не перемещая курсора, позицию, в которой заканчивается строка?

\$ — переход в конец строки

11. Выполните анализ опций редактора vi (сколько их, как узнать их назначение и т.д.).

Опции редактора vi позволяют настроить рабочую среду. Для задания опций используется команда set (в режиме последней строки): — : set all — вывести полный список опций; — : set nu — вывести номера строк; — : set list — вывести невидимые символы; — : set ic — не учитывать при поиске, является ли символ прописным или строчным.

12. Как определить режим работы редактора vi?

В редакторе vi есть два основных режима: командный режим и режим вставки. По умолчанию работа начинается в командном режиме. В режиме вставки клавиатура используется для набора текста. Для выхода в командный режим используется клавиша Esc или комбинация Ctrl + c.

13. Постройте граф взаимосвязи режимов работы редактора vi.(рис. fig. 6.1).

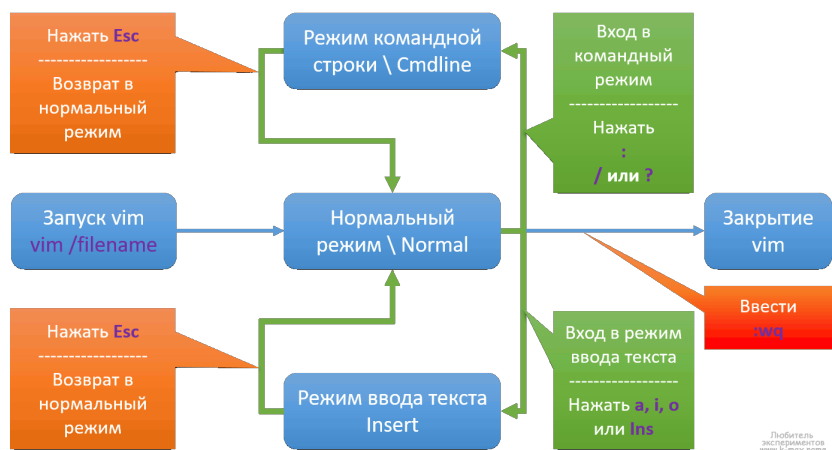


Рис. 6.1: Граф взаимосвязи режимов работы редактора vi