

Отчёт по лабораторной работе №8

Текстовой редактор vi

Желдакова Виктория Алексеевна

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
3.1	Создание нового файла с использованием vi	7
3.2	Редактирование существующего файла	9
4	Выводы	13
5	Контрольные вопросы	14

Список иллюстраций

3.1	Создание каталога, переход в него и вызов vi с созданием файла .	7
3.2	Окно vi после ввода текста и перехода в командный режим	8
3.3	Команда для сохранения изменений и завершения работы редактора	8
3.4	Изменение прав доступа файла	9
3.5	Вызов vi на редактирование файла	9
3.6	Результат редактирования в режиме вставки	9
3.7	Результат удаления слова, используя команду dw	10
3.8	Результат ввода слова в режиме вставки	10
3.9	Результат копирования и вставки строки	10
3.10	Результат удаления строки в командном режиме	11
3.11	Результат отмены последнего действия	11
3.12	Запись изменений и выход из vi	12
5.1	Граф взаимосвязи режимов работы редактора vi	17

Список таблиц

1 Цель работы

- Познакомиться с операционной системой Linux.
- Получить практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.

2 Теоретическое введение

`vi` — серия текстовых редакторов операционных систем семейства UNIX.

Первая версия была написана Биллом Джоем в 1976 году.

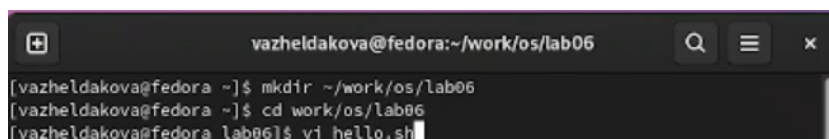
В то время наиболее распространённым был редактор `ed`. Поскольку он был довольно сложным для «простого смертного», George Coulouris разработал редактор `em` (англ. editor for mortals — «редактор для смертных»). Билл Джой модифицировал редактор `em` и назвал его `en`, а позже на его основе создал `ex`, в котором появился визуальный режим, вызывавшийся командой `vi`. Так как пользователи больше времени проводили в визуальном режиме, `ex 2.0`, ставший частью 2BSD, сразу запускался уже в нём. Так появился `vi`, бывший в то время всего лишь жёсткой ссылкой на `ex`.

Редактор писался для терминала ADM-3A, имевшего ограниченную клавиатуру: на ней не было выделенных клавиш-стрелок, а из модификаторов доступны только Shift и частично Ctrl. Эти обстоятельства повлияли на выбор используемых в редакторе клавиш.

3 Выполнение лабораторной работы

3.1 Создание нового файла с использованием vi

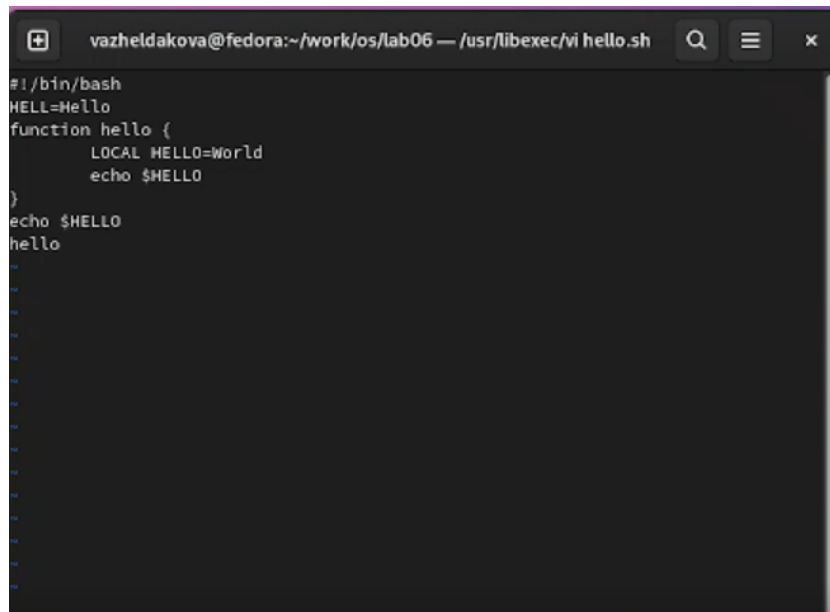
С помощью команды `mkdir` создали каталог `~/work/os/lab06`, перешли в него, используя `cd`, вызвали `vi` и создали новый файл `hello.sh` (рис. 3.1).

A screenshot of a terminal window with a dark background. The title bar at the top shows the username 'vazheldakova@fedora' and the current directory '~ /work/os/lab06'. The terminal contains three lines of commands and their outputs: the first line shows the execution of 'mkdir ~/work/os/lab06', the second line shows 'cd work/os/lab06', and the third line shows 'vi hello.sh' with a cursor at the end of the command.

```
vazheldakova@fedora: ~/work/os/lab06
[vazheldakova@fedora ~]$ mkdir ~/work/os/lab06
[vazheldakova@fedora ~]$ cd work/os/lab06
[vazheldakova@fedora lab06]$ vi hello.sh
```

Рис. 3.1: Создание каталога, переход в него и вызов `vi` с созданием файла

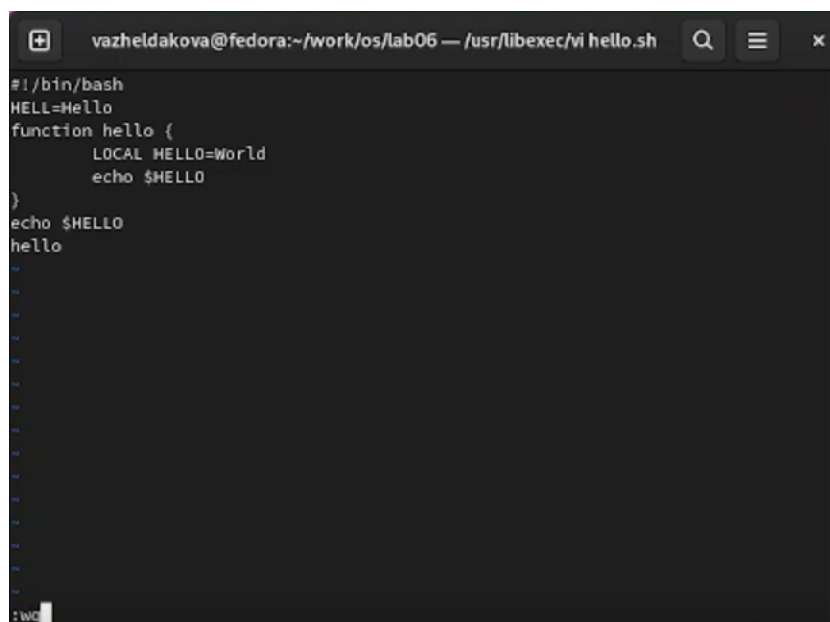
Нажали клавишу `i`, ввели приведённый в задании текст и перешли в командный режим, используя `Esc` (рис. 3.2).



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

Рис. 3.2: Окно vi после ввода текста и перехода в командный режим

Нажали двоеточие, ввели w и q, а затем нажали Enter для сохранения текста и завершения работы (рис. 3.3).



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

Рис. 3.3: Команда для сохранения изменений и завершения работы редактора

Использовали команду chmod для того, чтобы сделать файл исполняемым (рис. 3.4).


```
[vazheldakova@fedora lab06]$ chmod +x hello.sh
```

Рис. 3.4: Изменение прав доступа файла

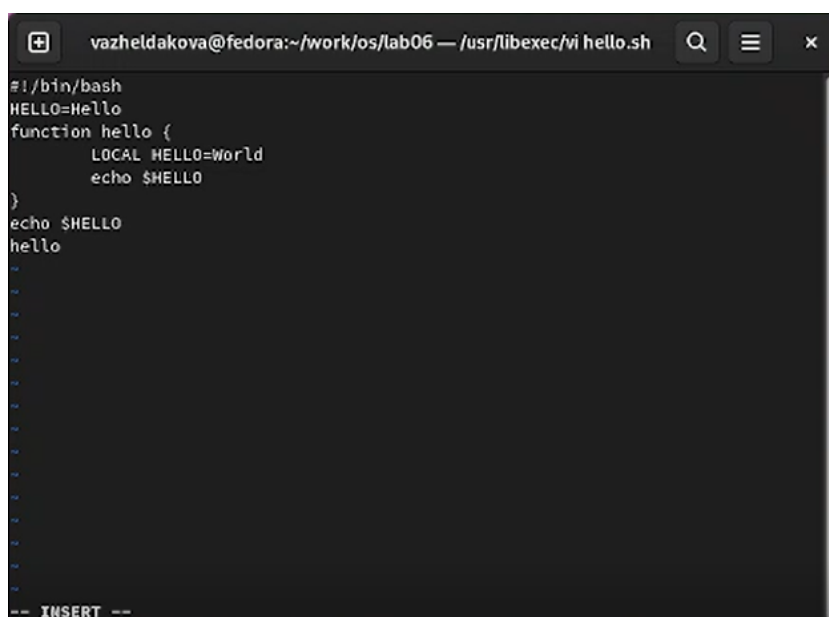
3.2 Редактирование существующего файла

Вызвали vi на редактирование файла (рис. 3.5).

```
[vazheldakova@fedora lab06]$ vi hello.sh
```

Рис. 3.5: Вызов vi на редактирование файла

Установили курсор в конец слова HELL, в режиме вставки заменили его на HELLO (рис. 3.6).



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

Рис. 3.6: Результат редактирования в режиме вставки

Нажали Esc для возврата в командный режим и с помощью команды dw удалили слово local (рис. 3.7).



```
vazheldakova@fedora:~/work/os/lab06 — /usr/libexec/vi hello.sh
#!/bin/bash
HELLO=Hello
function hello {
    echo $HELLO
}
echo $HELLO
hello
```

Рис. 3.7: Результат удаления слова, используя команду dw

Вернулись в режим вставки и написали слово LOCAL (рис. 3.8).



```
vazheldakova@fedora:~/work/os/lab06 — /usr/libexec/vi hello.sh
#!/bin/bash
HELLO=Hello
function hello {
    local=world
    echo $HELLO
}
echo $HELLO
hello
```

Рис. 3.8: Результат ввода слова в режиме вставки


Перешли в командный режим, с помощью команды Y скопировали в буфер строку echo \$HELLO, а затем вставили её в конец файла, используя P (рис. 3.9).



```
vazheldakova@fedora:~/work/os/lab06 — /usr/libexec/vi hello.sh
#!/bin/bash
HELLO=Hello
function hello {
    local=world
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO
```

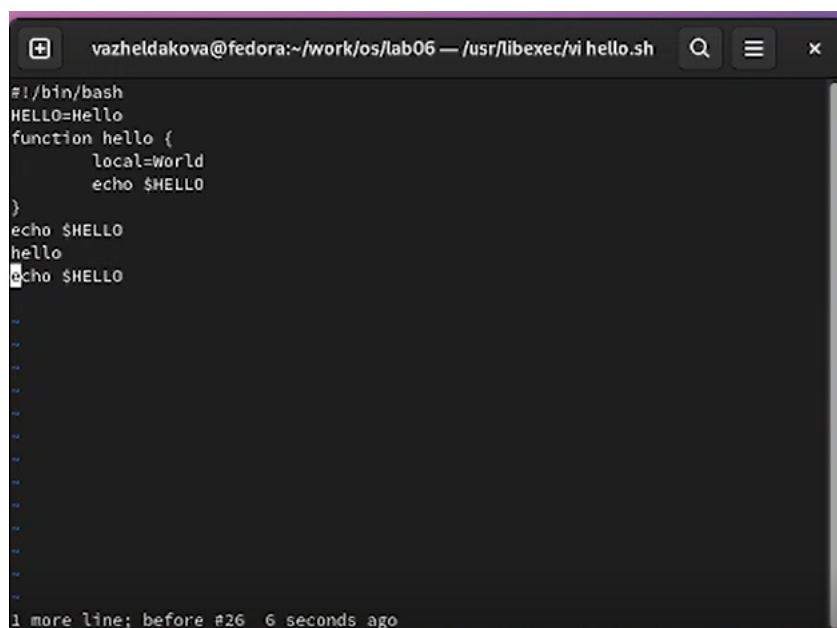
Рис. 3.9: Результат копирования и вставки строки

Используя dd, удалили только что вставленную строку и вернули её обратно с помощью команды u (рис. 3.10 и рис. 3.11).



```
vazheldakova@fedora:~/work/os/lab06 — /usr/libexec/vi hello.sh
#!/bin/bash
HELLO=Hello
function hello {
    local=world
    echo $HELLO
}
echo $HELLO
hello
```

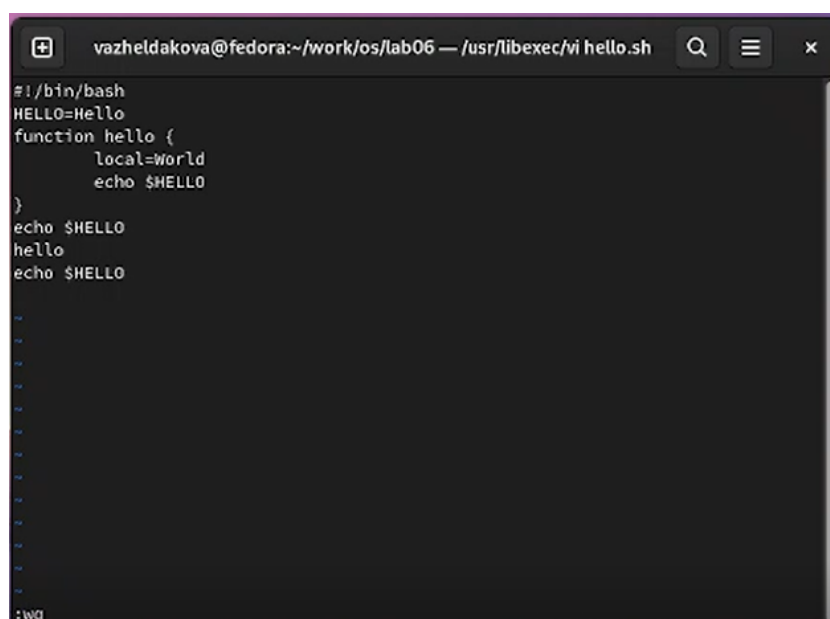
Рис. 3.10: Результат удаления строки в командном режиме



```
vazheldakova@fedora:~/work/os/lab06 — /usr/libexec/vi hello.sh
#!/bin/bash
HELLO=Hello
function hello {
    local=world
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO
1 more line; before #26 6 seconds ago
```

Рис. 3.11: Результат отмены последнего действия

Ввели двоеточие, w и q для записи изменений и выхода из vi (рис. 3.12).



```
vazheldakova@fedora:~/work/os/lab06 — /usr/libexec/vi hello.sh
#!/bin/bash
HELLO=Hello
function hello {
    local=World
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO

:~
```

Рис. 3.12: Запись изменений и выход из vi

4 Выводы

- Познакомились с операционной системой Linux.
- Получили практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.

5 Контрольные вопросы

1. Дайте краткую характеристику режимам работы редактора vi.

Командный режим позволяет управлять курсором и вводить команды редактирования. Режим вставки допускает производить ввод текста. При этом текст не будет восприниматься, как команды редактирования. Режим последней строки позволяет производить запись файла на диск и выходить из редактора Vi. Кроме того, используя этот режим, можно вводить дополнительные команды редактирования.

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

Если необходимо просто выйти Vi (без сохранения выполненных изменений), то необходимо в последней строке набрать символ q.

3. Назовите и дайте краткую характеристику командам позиционирования.

Команды позиционирования: w - перемещает курсор к началу следующего слова; nw - перемещает курсор к началу n-го слова; конец строки не останавливает движение курсора. Он продолжает подсчет слов с начала следующей строки; W - игнорирует все знаки пунктуации и перемещает курсор на следующее после пробела слово; e - перемещает курсор на последний символ в следующем слове; E - перемещает курсор на последний символ в слове. Игнорирует все символы пунктуации за исключением пробела. Слова разделяются пробелами; b - перемещает курсор к первому символу предыдущего слова; nb - перемещает курсор к первому символу n-го предыдущего слова.

Команда **b** не останавливается в начале строки и продолжает движение к началу строки; **B** - используется подобно команде **b**, за исключением того, что слова разделяются только символами пробел и новая строка. Она рассматривает все другие символы пунктуации как буквы.

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?

0 (ноль) - перейти в начало строки; **\$** - перейти в конец строки; **G** - перейти в конец файла; **nG** - перейти на строку номер **n**.

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

0 (ноль) - перейти в начало строки; **\$** - перейти в конец строки; **6.** Добавление/вставка текста **a**- добавить текст после курсора; **A** - добавить текст в конец строки; **i**- вставить текст перед курсором; **ni**- вставить текст **n** раз; **I** - вставить текст в началостроки. Вставка строки **o** - вставить строку под курсором; **O** - вставить строку над курсором. Удаление текста **x**- удалить один символ в буфер; **dw** - удалить одно слово в буфер; **d\$** - удалить в буфер текст от курсора до конца строки; **d0** - (ноль) - удалить в буфер текст от начала строки до позиции курсора; **dd** - удалить в буфер одну строку; **10dd**- удалить в буфер 10 строк. Отмена и повтор произведенных изменений **u**- отменить последнее изменение; - повторить последнее изменение. Копирование текста в буфер **Y**- скопировать строку в буфер; **nY**- скопировать **n** строк в буфер; **uw** - скопировать слово в буфер; Вставка текста из буфера **p**- вставить текст из буфера после курсора; **P**- вставить текст из буфера перед курсором. Замена текста **sw** - заменить слово; **n**

сw - заменить n слов; с\$ - заменить текст от курсора до конца строки; r - заменить слово; R - заменить текст. Поиск текста / - произвести поиск вперед по тексту указанной строки символов ; ? - произвести поиск назад по тексту указанной строки символов .

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

с\$ - заменить текст от курсора до конца строки.

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

u - отменить последнее изменение

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

Копирование и перемещение текста:

n,m d - уничтожить строки с n по m

Пример: : 3,8d

i,j m k - переместить строки с i по j , начиная со строки k

Пример : : 4,9 m 12

i,j t k- копировать строки с i по j на строку k

Пример: : 2,5 t 13

i,j w - записать строки с i по j в файл с именем

Пример: : 5,9 .

Запись в файл и выход из редактора:

w - записать измененный текст в файл на диске, не выходя из Vi;

:w - записать измененный текст в новый файл с именем;

:w! - записать измененный текст в файл с именем;

:wq- записать изменения в файл и выйти из Vi;

:q - выйти из редактора Vi;

:q!- выйти из редактора без записи;

:e!- вернуться в командный режим, отменив все изменения, произведенные со времени последней записи

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

\$ - перемещает курсор 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 режиме вставки внизу окна отобразится слово insert, в режиме последней строки - двоеточие, в командном режиме не отображается ничего.

13. Постройте граф взаимосвязи режимов работы редактора vi.

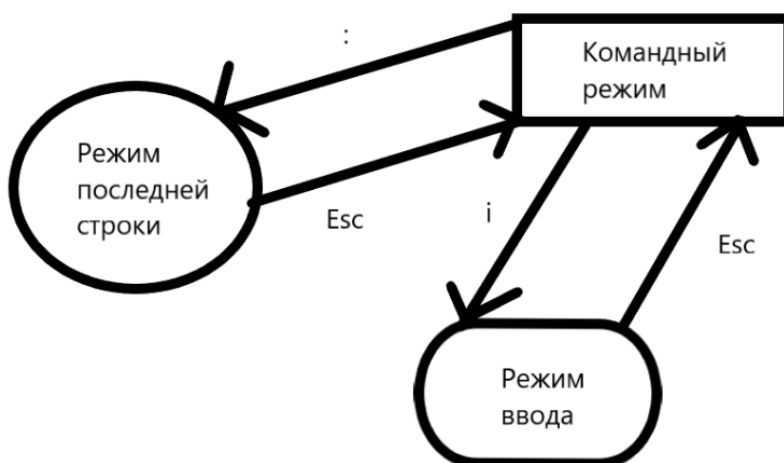


Рис. 5.1: Граф взаимосвязи режимов работы редактора vi