

# **Отчет по лабораторной работе №10**

**Дисциплина: Операционные системы**

Иванов Сергей Владимирович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>4</b>	<b>Контрольные вопросы</b>	<b>12</b>
<b>5</b>	<b>Вывод</b>	<b>16</b>

## Список иллюстраций

3.1	Создание файла через vi . . . . .	6
3.2	Запись в файл . . . . .	6
3.3	Режим последней строки . . . . .	7
3.4	Сохранение и выход . . . . .	7
3.5	Обновление прав . . . . .	7
3.6	Открытие с помощью текстового редактора . . . . .	8
3.7	Редактирование файла . . . . .	8
3.8	Редактирование файла . . . . .	9
3.9	Редактирование файла . . . . .	9
3.10	Редактирование файла . . . . .	10
3.11	Отмена последнего действия . . . . .	10
3.12	Сохранение и выход . . . . .	11
4.1	Граф взаимосвязи режимов работы редактора vi . . . . .	15

# 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 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. 1).

```
[svivanov1@svivanov1 ~]$ mkdir -p work3/os/lab06  
[svivanov1@svivanov1 ~]$ ls work/  
blog os SergeyIvanov21.github.io study  
[svivanov1@svivanov1 ~]$ cd work/os/lab06/  
[svivanov1@svivanov1 lab06]$ vi hello.sh
```

Рис. 3.1: Создание файла через `vi`

Нажимаю `i`, чтобы начать редактирование и добавлять текст (режим добавления), далее добавляю текст (рис. 2).

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

Рис. 3.2: Запись в файл

Нажимаю `esc`, чтобы выйти из режима добавления и перейти в командный режим. Перехожу в режим последней строки (рис. 3).



Рис. 3.3: 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. 4).



Рис. 3.4: Сохранение и выход

С помощью `chmod` добавляю права на исполнение файла, делая его исполняемым (рис. 5).

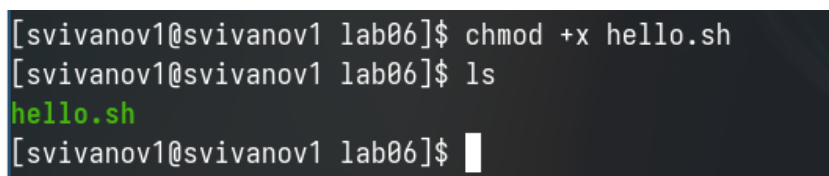
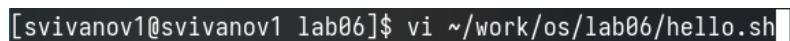


Рис. 3.5: Обновление прав

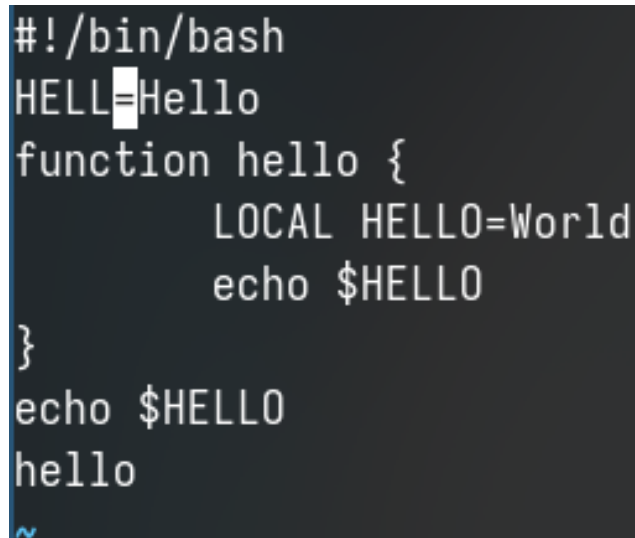
Снова открываю этот файл с помощью текстового редактора vi (рис. 6).



```
[svivanov1@svivanov1 lab06]$ vi ~/work/os/lab06/hello.sh
```

Рис. 3.6: Открытие с помощью текстового редактора

Изменяю местоположение курсора (рис. 7).



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

Рис. 3.7: Редактирование файла

В режиме добавления (клавиша i) удаляю слово LOCAL, снова перехожу в командный режим с помощью esc (рис. 8).



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

Рис. 3.8: Редактирование файла

В режиме добавления (клавиша i) добавляю слово local, снова перехожу в командный режим с помощью esc (рис. 9).

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

Рис. 3.9: Редактирование файла

В режиме добавления (клавиша i) перехожу в конец последней строки и добавляю новую строку, снова перехожу в командный режим с помощью esc. По-

сле этого возвращаюсь в режим добавления, удаляю строку и возвращаюсь в командный режим. (рис. 10)

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

Рис. 3.10: Редактирование файла

В командном режиме нажимаю “u” и отменяю последнее действие, удаление строки (рис. 11).

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

~

~

~

1 line less; before #3 32 seconds ago

Рис. 3.11: Отмена последнего действия

Перехожу в режим последней строки и ввожу w для сохранения файла и q для выхода, нажимаю enter и файл закрылся (рис. 12).

```
#!/bin/bash
HELL=Hello
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO
~
~
~
:wq
```

Рис. 3.12: Сохранение и выход

## 4 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?

С помощью `u` — 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. 4.1).

## vi: Режимы работы редактора

- **Командный режим** - нажатия клавиш интерпретируются как команды
  - исходным является режим, когда начинается сеанс vi редактирования
  - нажатие клавиши <Esc> переводит вас в командный режим
  - vi команды не отражаются на экране
  - ex команды префиксируются двоеточием(:) и отображаются на экране
- **Режим ввода** - информация, соответствующая нажатым клавишам вводится в файл

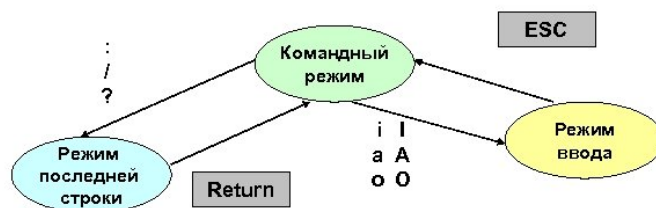


Рис. 4.1: Граф взаимосвязи режимов работы редактора vi

## 5 Вывод

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