

ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ №8

дисциплина: Операционные системы

Шилоносов Данил Вячеславович

Содержание

1	Цель работы	4
2	Задачи	5
3	Теоретическое введение	6
3.1	Указания к работе	6
3.2	Основные группы команд редактора	7
3.2.1	Команды позиционирования	7
3.2.2	Команды перемещения по файлу	7
3.2.3	Команды перемещения по словам ¹	7
3.3	Команды редактирования	7
3.3.1	Вставка текста	7
3.3.2	Вставка строки	8
3.3.3	Удаление текста	8
3.3.4	Отмена и повтор произведённых изменений	8
3.3.5	Копирование текста в буфер	8
3.3.6	Вставка текста из буфера	8
3.3.7	Замена текста	9
3.3.8	Поиск текста	9
3.4	Команды редактирования в режиме командной строки	9
3.4.1	Копирование и перемещение текста	9
3.4.2	Запись в файл и выход из редактора	9
3.5	Опции	10
4	Выполнение лабораторной работы	11
4.1	Задание 1. Создание нового файла с использованием vi	11
4.2	Задание 2. Редактирование существующего файла	12
5	Выводы	14

Список иллюстраций

4.1	Задание №1	12
4.2	Задание №2	13

1 Цель работы

Получение практических навыков работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.

2 Задачи

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

3 Теоретическое введение

3.1 Указания к работе

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.

Замечание. Следует помнить, что `vi` различает прописные и строчные буквы при наборе (восприятии) команд.

3.2 Основные группы команд редактора

3.2.1 Команды позиционирования

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

3.2.2 Команды перемещения по файлу

- Ctrl-d — перейти на пол-экрана вперёд;
- Ctrl-u — перейти на пол-экрана назад;
- Ctrl-f — перейти на страницу вперёд;
- Ctrl-b — перейти на страницу назад.

3.2.3 Команды перемещения по словам¹

- W или w — перейти на слово вперёд;
- n W или n w — перейти на n слов вперёд;
- b или B — перейти на слово назад;
- n b или n B — перейти на n слов назад.

3.3 Команды редактирования

3.3.1 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.

3.3.2 Вставка строки

- o — вставить строку под курсором;
- O — вставить строку над курсором.

3.3.3 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.

3.3.4 Отмена и повтор произведённых изменений

- u — отменить последнее изменение;
- . — повторить последнее изменение.

3.3.5 Копирование текста в буфер

- Y — скопировать строку в буфер;
- n Y — скопировать n строк в буфер;
- y w — скопировать слово в буфер.

3.3.6 Вставка текста из буфера

- p — вставить текст из буфера после курсора;
- P — вставить текст из буфера перед курсором.

3.3.7 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.

3.3.8 Поиск текста

- `/ текст` — произвести поиск вперёд по тексту указанной строки символов текст;
- `? текст` — произвести поиск назад по тексту указанной строки символов текст.

3.4 Команды редактирования в режиме командной строки

3.4.1 Копирование и перемещение текста

- `: n,m d` — удалить строки с `n` по `m`;
- `: i,j m k` — переместить строки с `i` по `j`, начиная со строки `k`;
- `: i,j t k` — копировать строки с `i` по `j` в строку `k`;
- `: i,j w имя-файла` — записать строки с `i` по `j` в файл с именем имя-файла.

3.4.2 Запись в файл и выход из редактора

- `: w` — записать изменённый текст в файл, не выходя из `vi`;
- `: w имя-файла` — записать изменённый текст в новый файл с именем имя-файла;
- `: w ! имя-файла` — записать изменённый текст в файл с именем имя-файла;
- `: w q` — 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, произведённые со времени последней записи.

3.5 Опции

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. Если вы хотите отказаться от использования опции, то в команде set перед именем опции надо поставить no.

4 Выполнение лабораторной работы

4.1 Задание 1. Создание нового файла с использованием vi

1. Создадим каталог с именем ~/work/os/lab06;
2. Перейдем во вновь созданный каталог;
3. Вызовем vi и создадим файл hello.sh;

```
vi hello.sh
```

4. Нажмем клавишу i и введем следующий текст

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

5. Нажмем клавишу Esc для перехода в командный режим после завершения ввода текста;
6. Нажмем : для перехода в режим последней строки и внизу нашего экрана появится приглашение в виде двоеточия;
7. Нажмем w (записать) и q (выйти), а затем нажмем клавишу Enter для сохранения нашего текста и завершения работы (рис. [4.1])

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

Рис. 4.1: Задание №1

8. Сделаем файл исполняемым и проверим изменение с помощью команд (рис. [4.2])

```
chmod +x hello.sh
```

```
ls -l
```

4.2 Задание 2. Редактирование существующего файла

1. Вызовите vi на редактирование файла

```
vi ~/work/os/lab08/hello.sh
```

2. Установим курсор в конец слова HELL второй строки;
3. Перейдем в режим вставки и заменим на HELLO. Нажмем Esc для возврата в командный режим;
4. Установим курсор на четвертую строку и сотрите слово LOCAL;
5. Перейдем в режим вставки и наберите следующий текст: local, нажмем Esc для возврата в командный режим;
6. Установим курсор на последней строке файла. Вставим после неё строку, содержащую следующий текст: echo \$HELLO;

7. Нажмем Esc для перехода в командный режим;
8. Удалим последнюю строку;
9. Введем команду отмены изменений и для отмены последней команды;
10. Введем символ : для перехода в режим последней строки. Запишем произведённые изменения и выйдем из vi. Итого (рис. [4.2])

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

Рис. 4.2: Задание №2

5 Выводы

В процессе выполнения лабораторной работы получены практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.