

# Лабораторная работа 10

## Текстовый редактор vi

Мочалкина Софья Васильевна

### Содержание

Цель работы .....	1
Задание .....	1
Теоретическое введение .....	2
Контрольные вопросы.....	9
Выводы .....	10

### Цель работы

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

### Задание

Задание 1. Создание нового файла с использованием vi

1) Создайте каталог с именем ~/work/os/lab06. 2) Перейдите во вновь созданный каталог. 3) Вызовите vi и создайте файл hello.sh vi hello.sh 4) Нажмите клавишу i и вводите следующий текст. 1 #!/bin/bash 2 HELL=Hello 3 function hello { 4 LOCAL HELLO=World 5 echo \$HELLO 6 } 7 echo \$HELLO 8 hello 5) Нажмите клавишу Esc для перехода в командный режим после завершения ввода текста. 6) Нажмите : для перехода в режим последней строки и внизу вашего экрана появится приглашение в виде двоеточия. 7) Нажмите w (записать) и q (выйти), а затем нажмите клавишу Enter для сохранения вашего текста и завершения работы. 8) Сделайте файл исполняемым 1 chmod +x hello.sh

Задание 2. Редактирование существующего файла 1) Вызовите vi на редактирование файла 1 vi ~/work/os/lab06/hello.sh 2) Установите курсор в конец слова HELL второй строки. 3) Перейдите в режим вставки и замените на HELLO. Нажмите Esc для возврата в командный режим. 4) Установите курсор на четвертую строку и сотрите слово LOCAL. 5) Перейдите в режим вставки и наберите следующий текст: local, нажмите Esc для возврата в командный режим. 6) Установите курсор на последней

строке файла. Вставьте после неё строку, содержащую следующий текст: echo \$HELLO. 7) Нажмите Esc для перехода в командный режим. 8) Удалите последнюю строку. 9) Введите команду отмены изменений u для отмены последней команды. 10) Введите символ : для перехода в режим последней строки. Запишите произведённые изменения и выйдите из vi.

## Теоретическое введение

8.2.1.2. Команды позиционирования – 0 (ноль) — переход в начало строки; – \$ — переход в конец строки; – G — переход в конец файла; –  $\square$  G — переход на строку с номером  $\square$ .

8.2.1.3. 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.

8.2.1.4. Команды перемещения по словам1 – W или w — перейти на слово вперёд; –  $\square$  W или  $\square$  w — перейти на  $\square$  слов вперёд; – b или B — перейти на слово назад; –  $\square$  b или  $\square$  B — перейти на  $\square$  слов назад.

### 8.2.2. Команды редактирования

8.2.2.1. Вставка текста – a — вставить текст после курсора; – A — вставить текст в конец строки; – i — вставить текст перед курсором; –  $\square$  i — вставить текст  $\square$  раз; – I — вставить текст в начало строки.

8.2.2.2. Вставка строки – o — вставить строку под курсором; – O — вставить строку над курсором.

8.2.2.3. 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; –  $\square$  d d — удалить в буфер  $\square$  строк.

8.2.2.4. Отмена и повтор произведённых изменений – u — отменить последнее изменение; – . — повторить последнее изменение.

8.2.2.5. Копирование текста в буфер – Y — скопировать строку в буфер; –  $\square$  Y — скопировать  $\square$  строк в буфер; – y w — скопировать слово в буфер.

8.2.2.6. Вставка текста из буфера – p — вставить текст из буфера после курсора; – P — вставить текст из буфера перед курсором.

8.2.2.7. Замена текста – c w — заменить слово; –  $\square$  c w — заменить  $\square$  слов; – c \$ — заменить текст от курсора до конца строки; – r — заменить слово; – R — заменить текст.

8.2.2.8. Поиск текста – / текст — произвести поиск вперёд по тексту указанной строки символов текст; – ? текст — произвести поиск назад по тексту указанной строки символов текст.

### 8.2.3. Команды редактирования в режиме командной строки

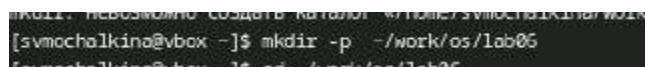
8.2.3.1. Копирование и перемещение текста – : 2,2 d — удалить строки с 2 по 2; – : 2,2 m 2 — переместить строки с 2 по 2, начиная со строки 2; – : 2,2 t 2 — копировать строки с 2 по 2 в строку 2; – : 2,2 w имя-файла — записать строки с 2 по 2 в файл с именем имя-файла.

8.2.3.2. 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.

8.2.4. 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. # Выполнение лабораторной работы

Задание 1. Создание нового файла с использованием vi

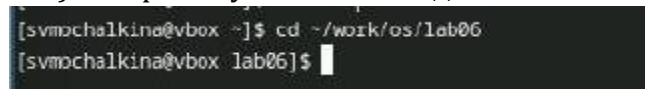
1)Создаю каталог с именем ~/work/os/lab06.



```
[svmochalkina@vbox ~]$ mkdir -p ~/work/os/lab06
```

рис.1

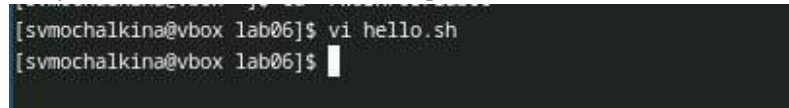
2) Перехожу во вновь созданный каталог.



```
[svmochalkina@vbox ~]$ cd ~/work/os/lab06
```

рис.2

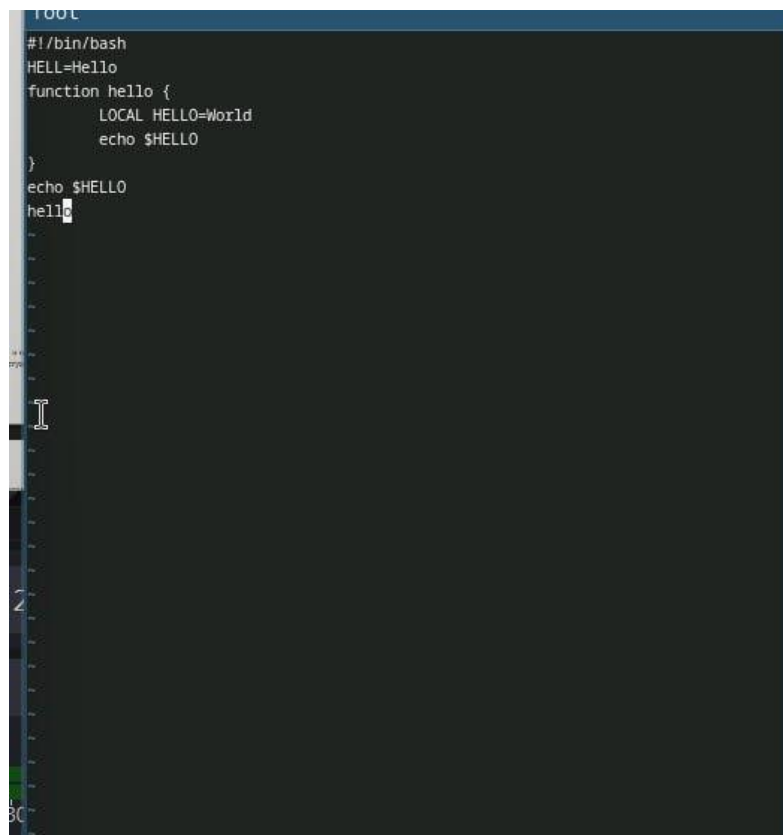
3) Вызываю vi и создаю файл hello.sh



```
[svmochalkina@vbox lab06]$ vi hello.sh
```

рис.3

4) Нажимаю клавишу i и ввожу следующий текст. 1 #!/bin/bash 2 HELLO=Hello 3 function hello { 4 LOCAL HELLO=World 5 echo \$HELLO 6 } 7 echo \$HELLO 8 hello

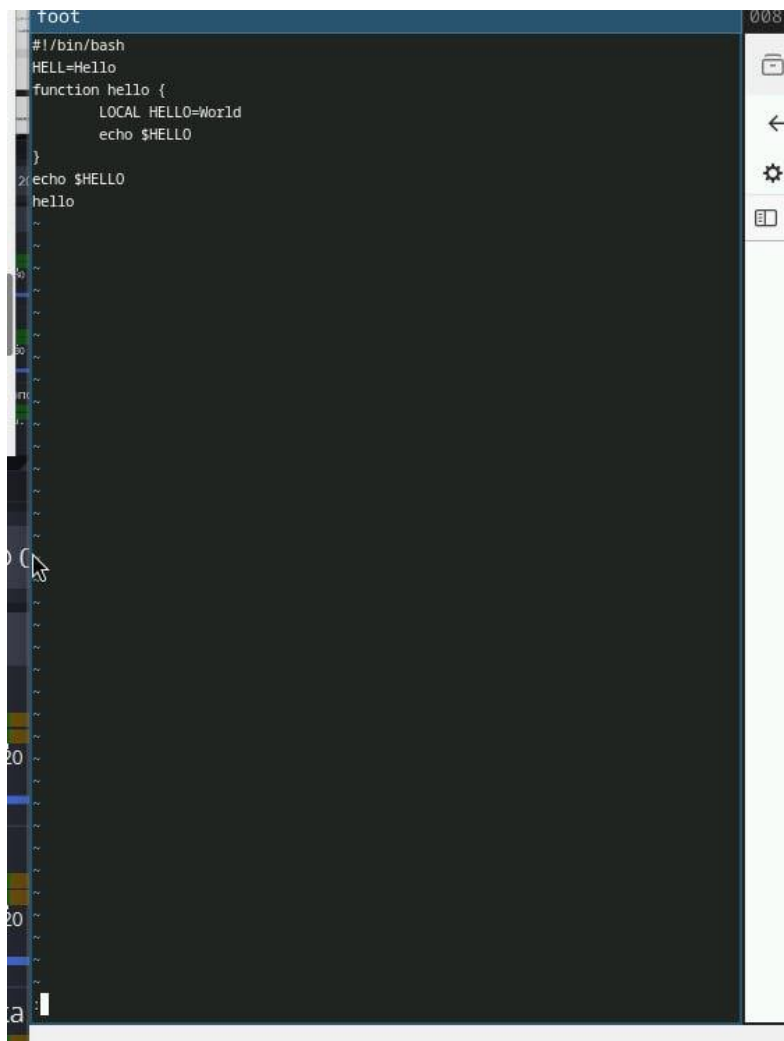


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
```

The screenshot shows a terminal window with a dark background and a light blue title bar. The terminal displays a shell script that sets a variable `HELL=Hello`, defines a function `hello` that sets a local variable `LOCAL HELLO=World` and echoes `$HELLO`, and then calls `echo $HELLO` and `hello`. The cursor is positioned at the end of the `hello` command. On the left side of the terminal, there is a vertical scrollbar and a status bar showing the file name `1001`.

рис.4

- 5) Нажмаю клавишу Esc для перехода в командный режим после завершения ввода текста.



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

рис.5

- 6) Нажимаю : для перехода в режим последней строки и внизу экрана появилось приглашение в виде двоеточия. 7) Нажимаю w (записать) и q (выйти), а затем нажимаю клавишу Enter для сохранения текста и завершения работы.

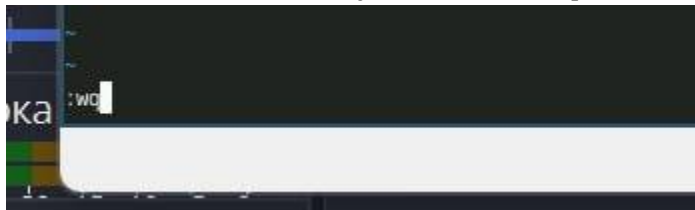


рис.6

- 8) Делаю файл исполняемым.

```
[svmochalkina@vbox lab06]$ vi hello.sh
[svmochalkina@vbox lab06]$ chmod +x hello.sh
[svmochalkina@vbox lab06]$
```

рис.7

Задание 2. Редактирование существующего файла 1) Вызываю vi на редактирование файла

```
[svmochalkina@vbox lab06]$ chmod +x hello.sh
[svmochalkina@vbox lab06]$ vi ~/work/os/lab06/hello.sh
[svmochalkina@vbox lab06]$
```

рис.8

2) Устанавливаю курсор в конец слова HELL второй строки.

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

рис.9

3) Перехожу в режим вставки и заменяю на HELLO. Нажимаю Esc для возврата в команд- ный режим.

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

рис.10

4) Устанавливаю курсор на четвертую строку и стираю слово LOCAL.

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

рис.11

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

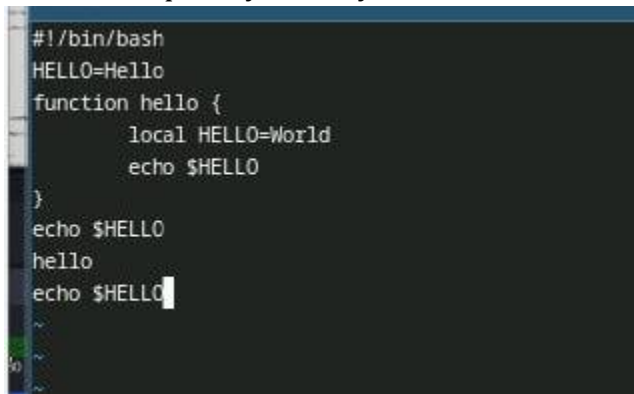
рис.12

- 5) Перехожу в режим вставки и набираю следующий текст: local, нажимаю Esc для возврата в командный режим.

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

рис.13

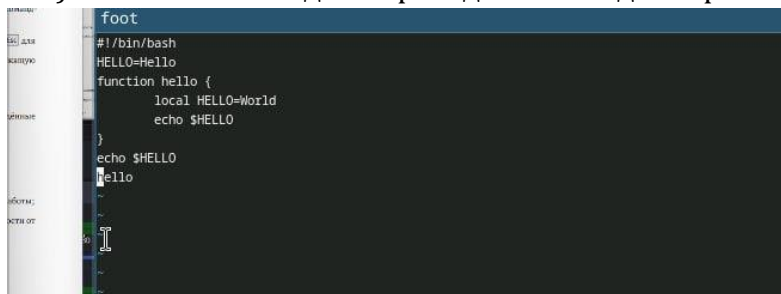
- 6) Устанавливаю курсор на последней строке файла. Вставляю после неё строку, содержащую следующий текст: echo \$HELLO.



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

рис.14

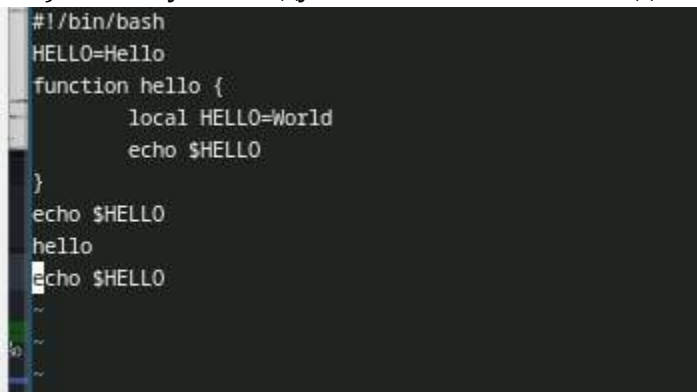
- 7) Нажимаю Esc для перехода в командный режим. 8) Удаляю последнюю строку.



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

рис.15

- 9) Ввожу команду отмены изменений и для отмены последней команды.



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

рис.16

- 10) Ввожу символ : для перехода в режим последней строки. Записываю произведённые изменения и выхожу из vi.



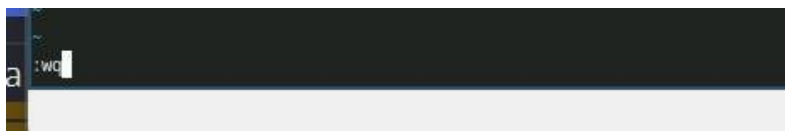


рис.17

## Контрольные вопросы

- 1) Режимы работы редактора vi Редактор vi работает в трех основных режимах:
  - Нормальный режим: Режим по умолчанию, в котором можно перемещаться по тексту и выполнять команды редактирования.
  - Режим вставки: Позволяет вводить текст. Вход в этот режим осуществляется с помощью команды i (вставка перед курсором) или a (вставка после курсора).
  - Режим последней строки: Позволяет вводить команды для сохранения, выхода и других операций. Вход в этот режим осуществляется с помощью клавиши :.
- 2) Как выйти из редактора, не сохраняя изменения? Чтобы выйти из редактора vi, не сохраняя изменения, введите :q! и нажмите Enter.
- 3) Команды позиционирования Команды позиционирования позволяют перемещаться по тексту:
  - h: перемещение влево на один символ.
  - j: перемещение вниз на одну строку.
  - k: перемещение вверх на одну строку.
  - l: перемещение вправо на один символ.
  - O: переход к началу строки.
  - \$: переход к концу строки.
  - G: переход к последней строке файла.
  - gg: переход к первой строке файла.
- 4) Что для редактора vi является словом? В редакторе vi словом считается последовательность символов, разделенная пробелами или знаками препинания. Например, слово может состоять из букв, цифр и некоторых специальных символов.
- 5) Как перейти в начало (конец) файла?
  - Для перехода в начало файла используйте gg.
  - Для перехода в конец файла используйте G.
- 6) Основные группы команд редактирования
  - Вставка: команды для добавления текста (i, a, o, O).
  - Удаление: команды для удаления текста (d, x, dw, dd).
  - Копирование и вставка: команды для копирования (y) и вставки (p).
  - Замена: команда для замены символов (r, R).

- 7) Заполнение строки символами \$ Чтобы заполнить строку символами \$, выполните следующие действия:
- 8) Перейдите в режим вставки, нажав i.
- 9) Введите символ \$ столько раз, сколько необходимо.
- 10) Нажмите Esc для выхода из режима вставки.
- 11) Как отменить некорректное действие? Чтобы отменить последнее действие, используйте команду u в нормальном режиме.
- 12) Основные группы команд режима последней строки
  - Сохранение: :w (сохранить изменения).
  - Выход: :q (выйти из редактора).
  - Сохранение и выход: :wq или :x.
  - Принудительный выход: :q! (выйти без сохранения).
- 10) Как определить позицию, в которой заканчивается строка? В нормальном режиме можно использовать команду \$, которая переместит курсор в конец текущей строки. Также можно посмотреть номер строки и позицию курсора в строке в правом нижнем углу экрана.
- 11) Опции редактора vi Опции редактора vi можно узнать с помощью команды :set. Некоторые из них:
  - number: отображать номера строк.
  - ignorecase: игнорировать регистр при поиске.
  - showmode: показывать текущий режим работы.
- 12) Как определить режим работы редактора vi? Режим работы редактора можно определить по тому, как реагирует курсор на нажатие клавиш. В нормальном режиме курсор перемещается, в режиме вставки — вводится текст, а в режиме последней строки вы можете вводить команды.
- 13) Граф взаимосвязи режимов работы редактора vi` Нормальный режим | | (вход в режим вставки) v Режим вставки ^ | | (вход в режим последней строки) | Режим последней строки

## Выводы

Я познакомилась с операционной системой Linux. Получила практические навыки работы с редактором vi, установленным по умолчанию практически во всех дистрибутивах.