

Отчет по лабораторной работе №9

Российский Университет Дружбы Народов

Факультет Физико-Математических и Естественных Наук

Дисциплина: *Операционные системы*

Студент: Алхатиб Осама

Группа: НПИбд-02-20

2021г.

Цель работы

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

Последовательность выполнения работы

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

Введение

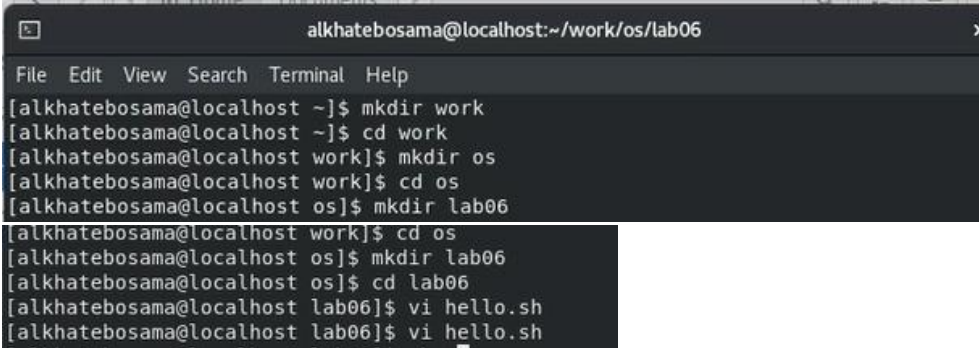
- режим последней (или командной) строки — используется для записи изменений в файл и выхода из редактора. Для вызова редактора vi необходимо указать команду vi и имя редактируемого файла: vi При этом в случае отсутствия файла с указанным именем будет создан такой файл. Переход в командный режим осуществляется 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.
- 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;
-

Ход работы:

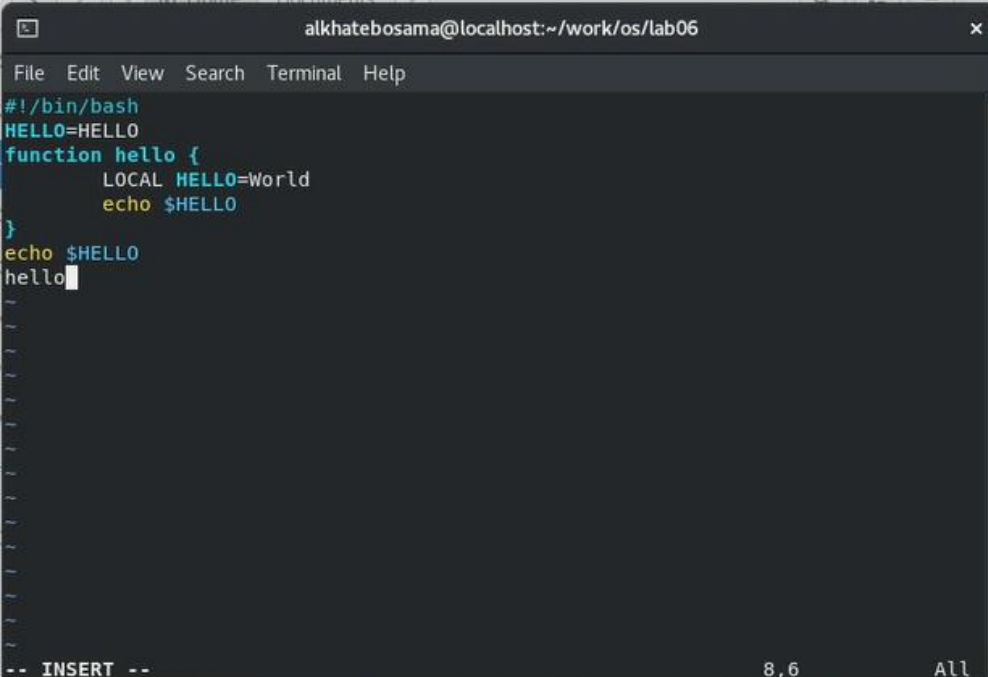
Задание 1. Создание нового файла с использованием vi.

1. Создал каталог с именем ~/work/os/lab06.
2. Перешел во вновь созданный каталог.
3. Вызвал vi и создал файл hello.sh



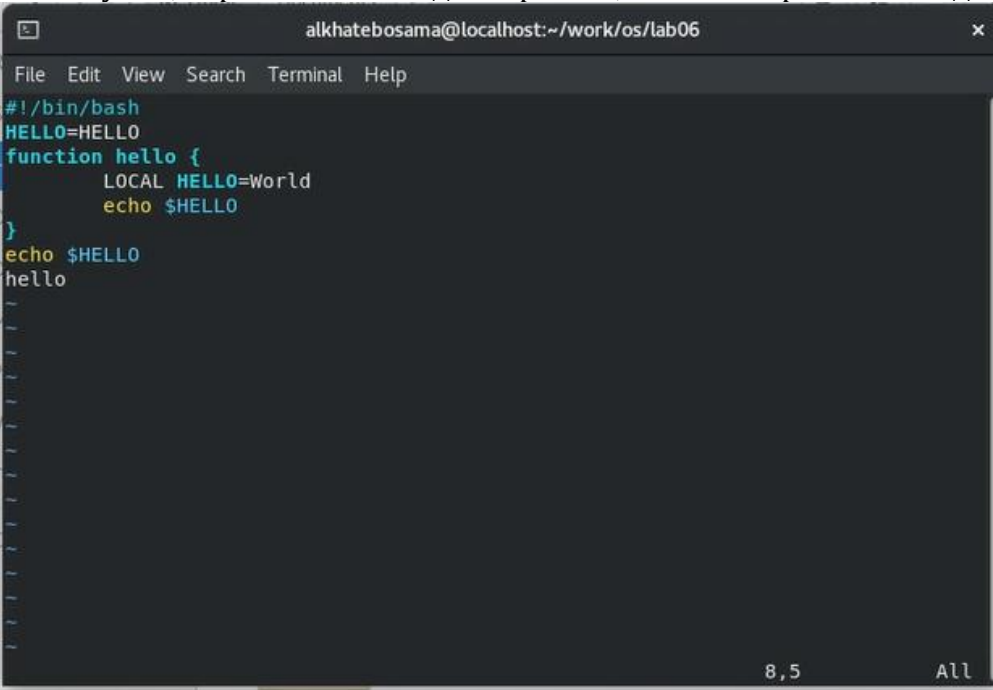
```
alkhatebosama@localhost:~/work/os/lab06
File Edit View Search Terminal Help
[alkhatebosama@localhost ~]$ mkdir work
[alkhatebosama@localhost ~]$ cd work
[alkhatebosama@localhost work]$ mkdir os
[alkhatebosama@localhost work]$ cd os
1. [alkhatebosama@localhost os]$ mkdir lab06
[alkhatebosama@localhost work]$ cd os
[alkhatebosama@localhost os]$ mkdir lab06
[alkhatebosama@localhost os]$ cd lab06
2. [alkhatebosama@localhost lab06]$ vi hello.sh
[alkhatebosama@localhost lab06]$ vi hello.sh
```

4. Нажав клавишу i, ввел следующий текст:



```
alkhatebosama@localhost:~/work/os/lab06
File Edit View Search Terminal Help
#!/bin/bash
HELLO=HELLO
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
-- INSERT -- 8,6 All
```

5. Нажав клавишу Esc, перешел в командный режим, после завершения ввода



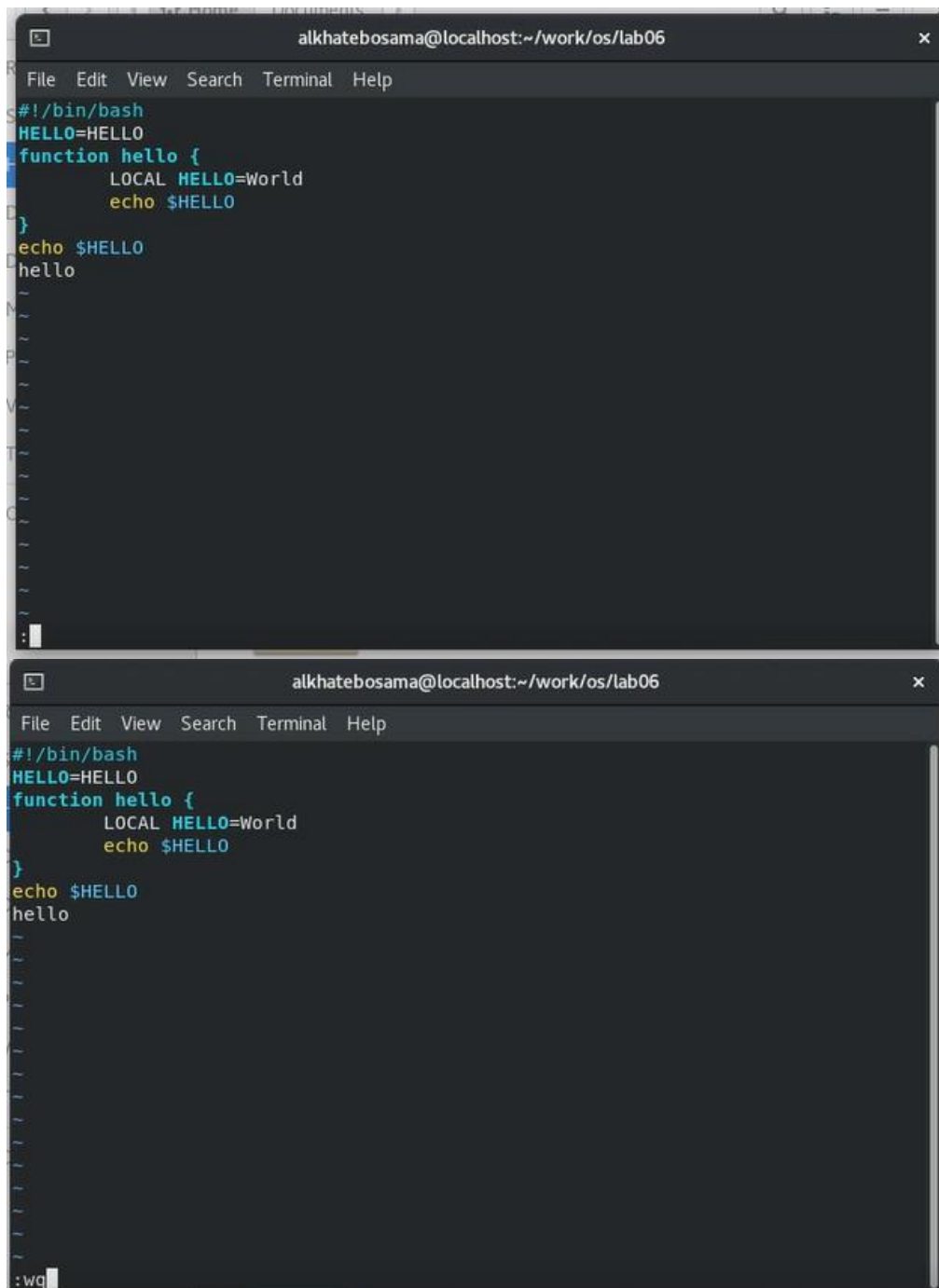
The screenshot shows a terminal window titled 'alkhatebosama@localhost:~/work/os/lab06'. The window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal content is as follows:

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

Below the script, there are several lines of tilde characters (~) representing a scrollable history. At the bottom right of the terminal, the text '8,5' and 'All' are visible.

текста.

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



The image shows two screenshots of a terminal window. The top screenshot shows the terminal with a menu bar (File, Edit, View, Search, Terminal, Help) and a dark background. The prompt is 'alkhatebosama@localhost:~/work/os/lab06'. The script content is:
#!/bin/bash
HELLO=HELLO
function hello {
 LOCAL HELLO=World
 echo \$HELLO
}
echo \$HELLO
hello
The bottom screenshot shows the same terminal after editing. The prompt is now ':wq' at the end of the line, indicating the file has been saved and the editor has exited.

8. Сделал файл исполняемым.

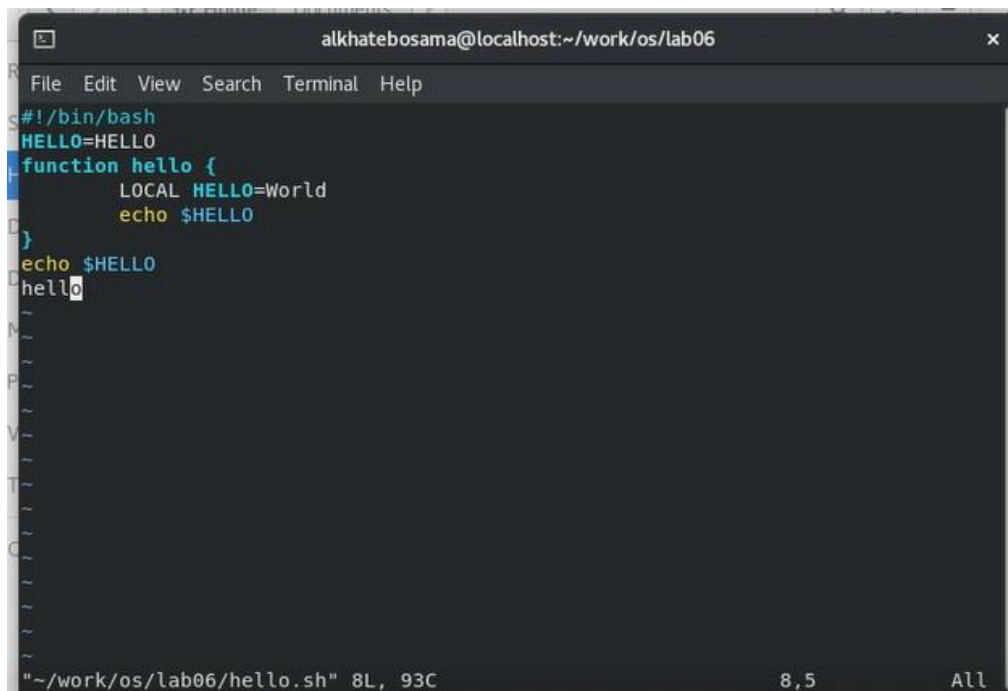
1. [alkhatebosama@localhost lab06]\$ vi hello.sh
[alkhatebosama@localhost lab06]\$ chmod +x hello.sh

1. Вызвал vi на редактирование файла. 1.

[alkhatebosama@localhost lab06]\$ chmod +x hello.sh
[alkhatebosama@localhost lab06]\$ vi ~/work/os/lab06/hello.sh

2. Установил

курсор в конец слова HELLO второй строки. 1.



```
alkhatebosama@localhost:~/work/os/lab06
File Edit View Search Terminal Help
#!/bin/bash
HELLO=HELLO
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
```

"~/work/os/lab06/hello.sh" 8L, 93C 8,5 All

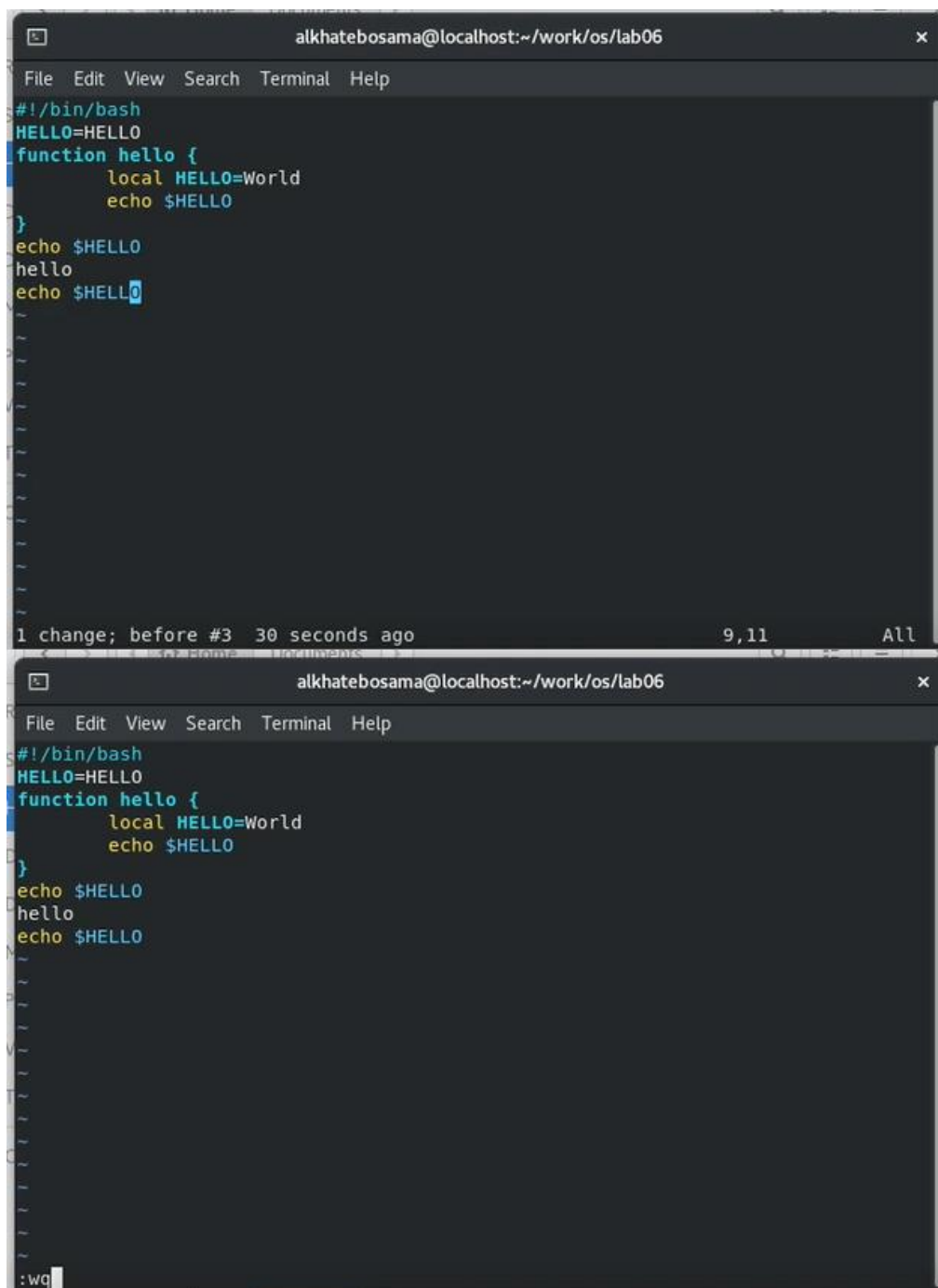
3. Перешел в режим вставки и заменил на HELLO. Нажал Esc для возврата в командный режим. 4. Установил курсор на четвертую строку и стёр слово LOCAL. 5. Перешел в режим вставки и набрал следующий текст: local, нажала Esc для возврата в командный режим. 6. Установил курсор на последней строке файла. Вставил после неё строку, содержащую следующий текст: echo \$HELLO. 7. Нажал Esc для перехода в командный режим. 8. Удалил последнюю строку. 9. Ввел команду отмены изменений и для отмены последней команды. 10. Ввел символ «:» для перехода в режим последней строки. Записал произведённые изменения и вышел из vi.


```
alkhatebosama@localhost:~/work/os/lab06
File Edit View Search Terminal Help
#!/bin/bash
HELLO=HELLO
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello
echo $HELLO

-- INSERT -- 9,12 All
```

```
alkhatebosama@localhost:~/work/os/lab06
File Edit View Search Terminal Help
#!/bin/bash
HELLO=HELLO
function hello {
    local HELLO=World
    echo $HELLO
}
echo $HELLO
hello

-- INSERT -- 9,1 All
```



The image shows two sequential screenshots of a terminal window. The window title is 'alkhatebosama@localhost:~/work/os/lab06'. The terminal has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The script content is as follows:

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

In the first screenshot, the cursor is at the end of the third line of the script. In the second screenshot, the script has been executed, resulting in the following output:

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

The output shows the global variable value 'HELLO' on the first line, the function output 'World' on the second line, and the global variable value 'HELLO' on the third line. The prompt ':wq' is visible at the bottom of the terminal.

Вывод

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

Ответы на контрольные вопросы:

1. Краткая характеристика режимов 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. Выйти из редактора, не сохраняя произведённые изменения, можно используя клавиши «:q!» в командном режиме. 3. Краткую характеристику команд 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. Для редактора vi словом является: пробел; буквы, находящиеся между двумя пробелами. 5. Из любого места редактируемого файла перейти в конец файла можно с помощью клавиши G и курсора вниз, а в начало — курсор вверх. 6. Краткая характеристика основных групп 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; * u 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. Чтобы заполнить строку символами \$ можно использовать клавиши n i (вставить текст n раз). 8. Отменить некорректное действие, связанное с процессом редактирования, можно с помощью клавиши «.» 9. Характеристика основных групп команд режима последней строки: 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. 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 имяфайла; * : w ! имя-файла — записать изменённый текст в файл с именем имяфайла; * : 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. 10. Определить, не перемещая курсора, позицию, в которой заканчивается строка, можно используя клавишу \$ (переход в конец строки). 11. 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. 12. Определить режим работы редактора vi можно по последней командной строке. 13. Взаимосвязь режимов работы редактора vi: