

Отчет по лабораторной работе №8

Операционные системы

Фадин В.В.

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Задание 1. Создание нового файла с использованием vi	6
2.2	Задание 2. Редактирование существующего файла	8
3	Выводы	11
4	Ответы на онтрольные вопросы	12
4.0.1	1. Режимы работы редактора vi	12
4.0.2	2. Выход из редактора, не сохраняя произведённые изменения.	12
4.0.3	3. Команды сказок	12
4.0.4	4. Слово в редакторе vi	13
4.1	начало# 5. Переход в (конец) файла	13
4.1.1	6. Основные группы команд редактирования	13
4.1.2	7. Заполнение строк символами \$	13
4.1.3	8. Отмена некорректного действия	13
4.1.4	9. Основная группа управления верхних строк.	14
4.1.5	10. Определение позиции, в которой заканчивается строка	14
4.1.6	11. Анализ опций редактора vi	14
4.1.7	12. Определение режима работы редактора vi	14
4.1.8	13. Граф взаимосвязи периодов работы редактора vi	14

Список иллюстраций

2.1	hello.sh	6
2.2	hello.sh	7
2.3	hello.sh	7
2.4	hello.sh	8
2.5	hello.sh	9
2.6	hello.sh	9
2.7	hello.sh	10
2.8	hello.sh	10

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 Выполнение лабораторной работы

2.1 Задание 1. Создание нового файла с использованием vi

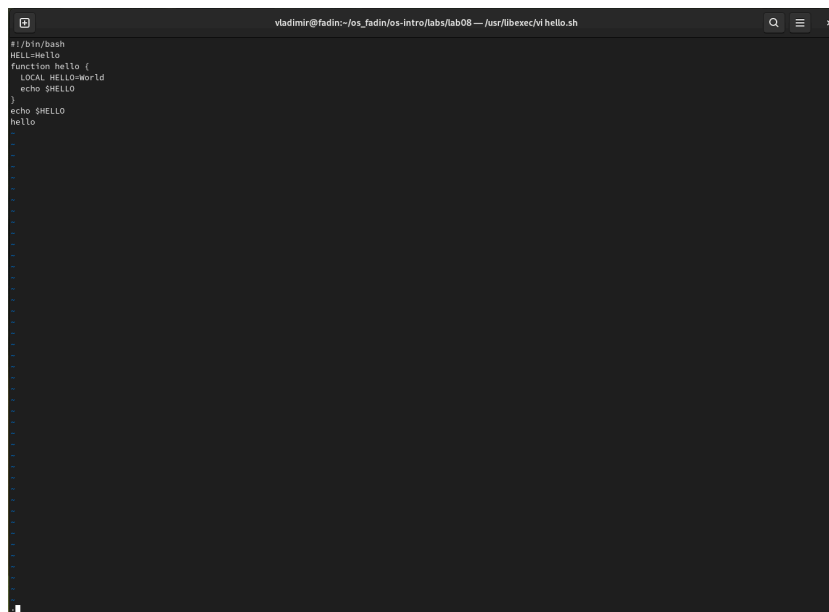
Перейдем в каталог с именем `~os_fadin/os-intro/labs/lab08/`.
Вызовем `vi` и создайте файл `hello.sh`, также введем текст скрипта.



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

Рис. 2.1: `hello.sh`

Нажмем клавишу `Esc` для перехода в командный режим после завершения ввода текста. Затем нажмем `:` для перехода в режим последней строки и внизу экрана появилась приглашение в виде двоеточия.

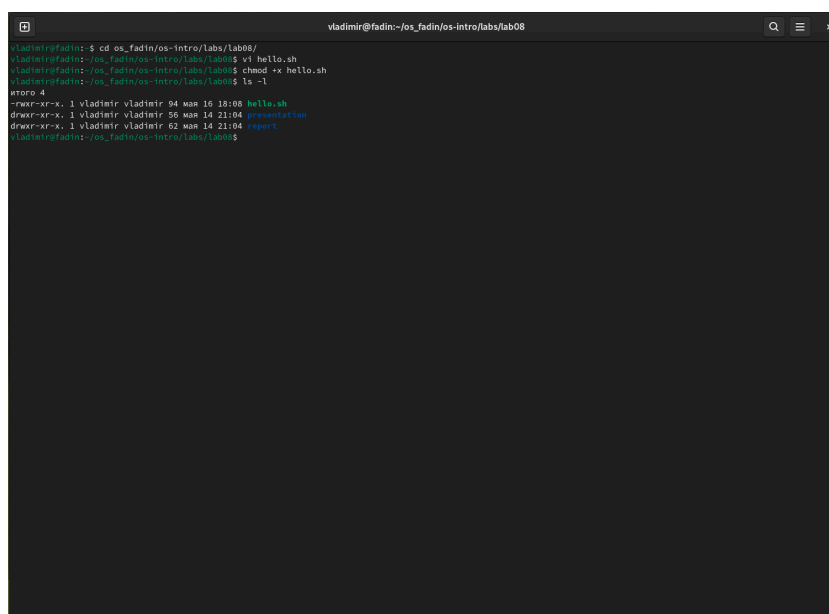


```
#!/bin/bash
HELLO=Hello
function hello {
    LOCAL HELLO=world
    echo $HELLO
}
echo $HELLO
hello
```

Рис. 2.2: hello.sh

Нажмем **w** (записать) и **q** (выйти), а затем нажмем клавишу **Enter** для сохранения текста и завершения работы.

Сделаем файл исполняемым.

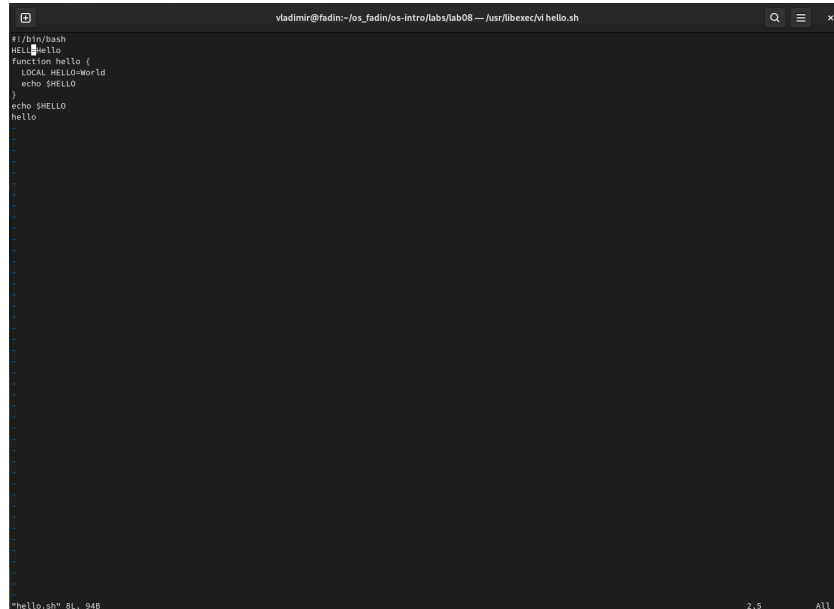


```
vladimir@fadin:~/os_fadin/os-intro/labs/lab08$
vladimir@fadin:~/os_fadin/os-intro/labs/lab08$ vi hello.sh
vladimir@fadin:~/os_fadin/os-intro/labs/lab08$ chmod +x hello.sh
vladimir@fadin:~/os_fadin/os-intro/labs/lab08$ ls -l
-rwxr-xr-x. 1 vladimir vladimir 94 was 16 18:08 hello.sh
-rwxr-xr-x. 1 vladimir vladimir 56 was 14 21:04 presentation
-rwxr-xr-x. 1 vladimir vladimir 63 was 14 21:04 report
vladimir@fadin:~/os_fadin/os-intro/labs/lab08$
```

Рис. 2.3: hello.sh

2.2 Задание 2. Редактирование существующего файла

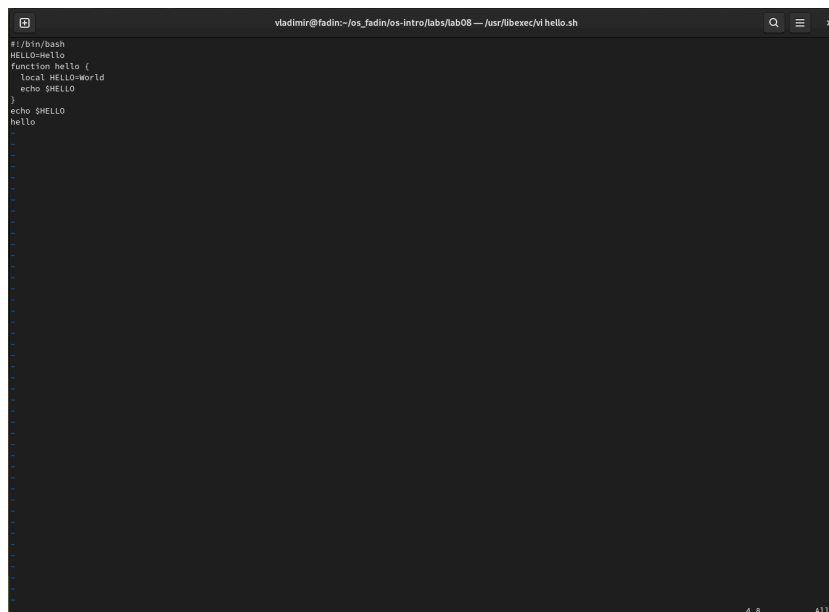
Вызовем `vi` на редактирование файла `hello.sh` и установим курсор в конец слова `HELL` второй строки.



```
vi /bin/bash
HELLhello
function hello {
  LOCAL HELLO=world
  echo $HELLO
}
echo $HELLO
hello
```

Рис. 2.4: `hello.sh`

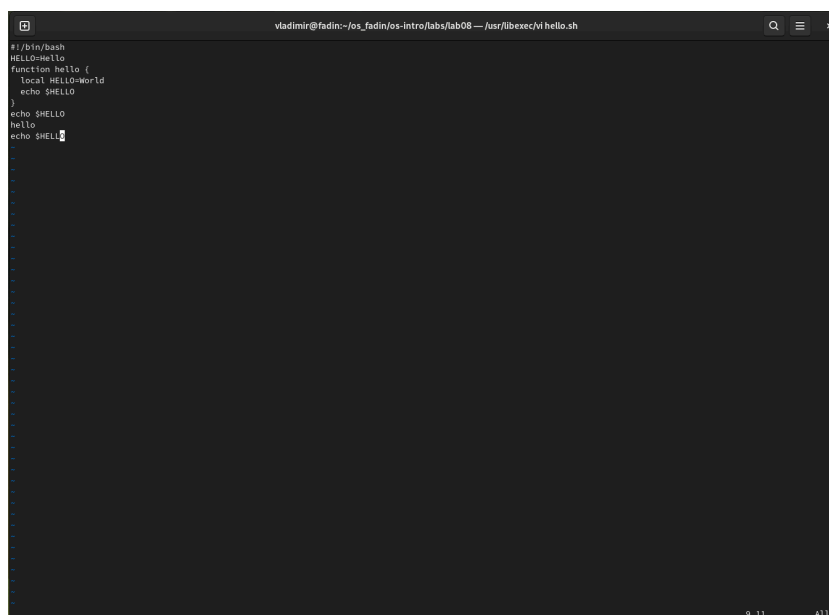
Перейдем в режим вставки и заменим на `HELLO`. Нажмем `Esc` для возврата в командный режим. Затем установим курсор на четвертую строку и сотрем слово `LOCAL`. После чего перейдем в режим вставки и наберем следующий текст: `local`, нажмем `Esc` для возврата в командный режим.



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

Рис. 2.5: hello.sh

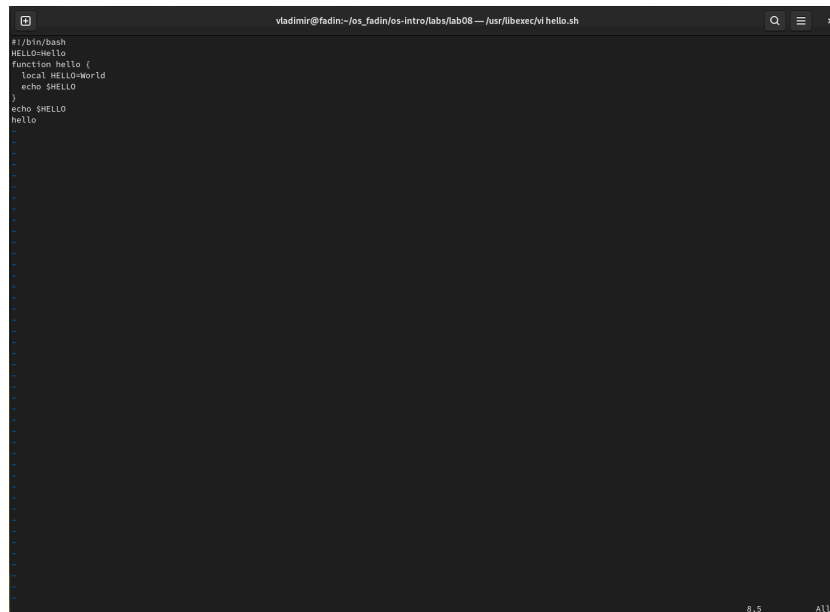
Установим курсор на последней строке файла. Вставим после неё строку, содержащую следующий текст: `echo $HELLO`. Затем нажмем Esc для перехода в командный режим.



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

Рис. 2.6: hello.sh

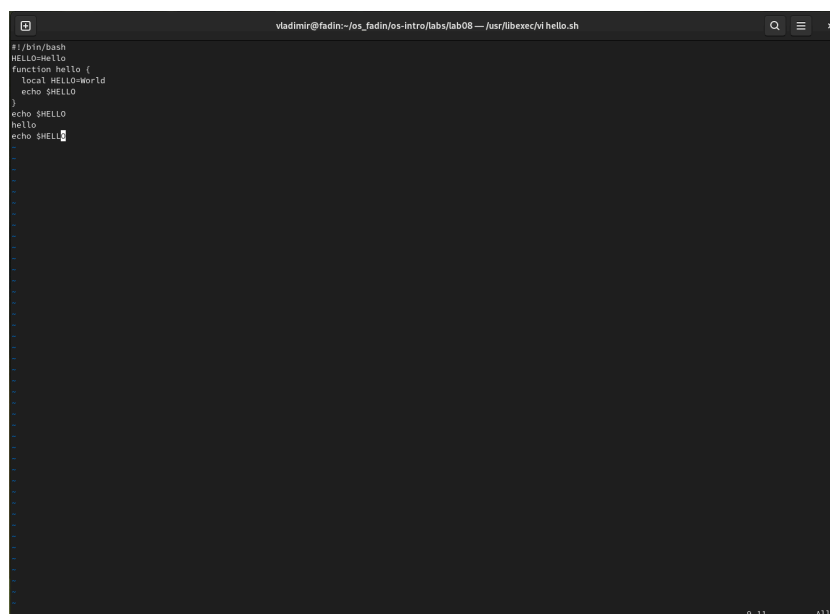
Удалим последнюю строку.



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

Рис. 2.7: hello.sh

Введем 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. 2.8: hello.sh

Введем символ : для перехода в режим последней строки. Запишем произведённые изменения и выйдите из vi.

3 Выводы

Познакомились с операционной системой Linux. Получили практические навыки работы с редактором vi.

4 Ответы на онтрольные вопросы

4.0.1 1. Режимы работы редактора vi

Редактор vi имеет несколько режимов работы:

- **Командный режим:** это режим по умолчанию, в котором вы можете вводить команды для навигации, редактирования и управления текстом.
- **Режим вставки:** этот режим позволяет вставлять текст в файл.
- **Режим строки:** этот режим позволяет редактировать одну строку текста.

4.0.2 2. Выход из редактора, не сохраняя произведённые изменения.

Чтобы выйти из редактора vi без сохранения изменений, нажмите :q, а затем Enter.

4.0.3 3. Команды сказок

Некоторые распространенные команды позиционирования в vi включают:

- h: перемещает курсор влево.
- j: перемещает курсор вниз.
- k: перемещает курсор вверх.
- l: перемещает курсор вправо.
- w: перемещает курсор в начало следующего слова.
- b: перемещает курсор в начало предыдущего слова.

- `0`: перемещает курсор в начало строки.
- `$`: перемещает курсор в конец строки.

4.0.4 4. Слово в редакторе vi

В vi слово представляет собой последовательность буквенно-цифровых символов, разделенных пробелами или знаками препинания.

4.1 начало# 5. Переход в (конец) файла

Чтобы перейти к началу файла, нажмите `gg`. Чтобы перейти к концу файла, нажмите `G`.

4.1.1 6. Основные группы команд редактирования

Некоторые основные команды редактирования в vi включают:

- `i`: переход в режим вставки.
- `a`: переход в режим вставки после курсора.
- `o`: открывает новую строку под курсором и переходит в режим вставки.
- `dw`: удаляет слово под курсором.
- `dd`: Удаляет строку под курсором.
- `p`: вставляет содержимое буфера обмена после курсора.

4.1.2 7. Заполнение строк символами \$

Чтобы заполнить строку символами `$`, нажмите `esc` для входа в командный режим, затем введите `:r` и нажмите `Enter`.

4.1.3 8. Отмена некорректного действия

Чтобы отменить неправильное действие, нажмите `u` в командном режиме.

4.1.4 9. Основная группа управления верхних строк.

Некоторые распространенные команды в режиме последней строки (также известном как режим `ex`) включают:

- `:w`: Сохраняет файл.
- `:q`: Выход из редактора.
- `:q`: Закрывает редактор без сохранения изменений.
- `:wq`: сохраняет файл и закрывает редактор.

4.1.5 10. Определение позиции, в которой заканчивается строка

Чтобы определить позицию, где заканчивается строка, не перемещая курсор, нажмите `«esc»` для входа в командный режим, затем введите `«:$»` и нажмите `«Enter»`. Это отобразит номер строки и номер столбца конца строки.

4.1.6 11. Анализ опций редактора `vi`

`Vi` имеет множество опций, которые можно настроить для настройки его поведения. Эти параметры можно установить с помощью команды `:set` в режиме `ex`. Например, `:set nu` устанавливает нумерацию строк, а `:set ai` устанавливает автоотступ.

4.1.7 12. Определение режима работы редактора `vi`

Чтобы определить текущий режим редактора `vi`, найдите индикатор режима внизу экрана. В командном режиме индикатор отобразит `--`, а в режиме вставки индикатор отобразит `-- INSERT --`.

4.1.8 13. Граф взаимосвязи периодов работы редактора `vi`

График режимов `vi` выглядит следующим образом:

- Командный режим > Режим вставки (с использованием i, a или o)
- Режим вставки > Режим команд (с помощью esc)
- Командный режим > Режим последней строки (с использованием :)
- Режим последней строки > Командный режим (с помощью Enter)

Обратите внимание, что этот график не является исчерпывающим, но показывает основные переходы между режимами в vi.