

Лабораторная работа 7

Текстовый редактор vi

Генералов Даниил, НПИ-01-21, 1032212280

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Контрольные вопросы	11
6	Выводы	15

Список иллюстраций

4.1	Окно после ввода текста	8
4.2	Делаем файл исполняемым	9
4.3	Исправленный текст	9
4.4	Запускаем файл	10

Список таблиц

1 Цель работы

Целью данной работы является:

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

2 Задание

Требуется сделать следующее:

- Ввести текст в редактор и сохранить его;
- Сделать файл с текстом исполняемым;
- Исправить ошибки в коде.

3 Теоретическое введение

VI – это текстовый редактор с полноэкранным визуальным интерфейсом. Как замечено в задании, он присутствует в той или иной форме на большинстве Unix-подобных систем, включая настольные и серверные дистрибутивы Linux, Mac OS X/macOS, а также иногда на встраиваемых системах вроде роутеров. Из-за своей распространенности, а также из-за его возможностей, полезно знать как именно использовать его для ввода текста.

4 Выполнение лабораторной работы

Сначала нужно создать текстовый документ с указанным содержимым. Для этого нужно сначала запустить редактор, затем перевести его в режим редактирования нажатием клавиши `i`. После этого можно ввести текст, и когда ввод закончен, можно закрыть режим редактирования клавишей `Esc`. Состояние окна после этого видно на рис. 4.1.

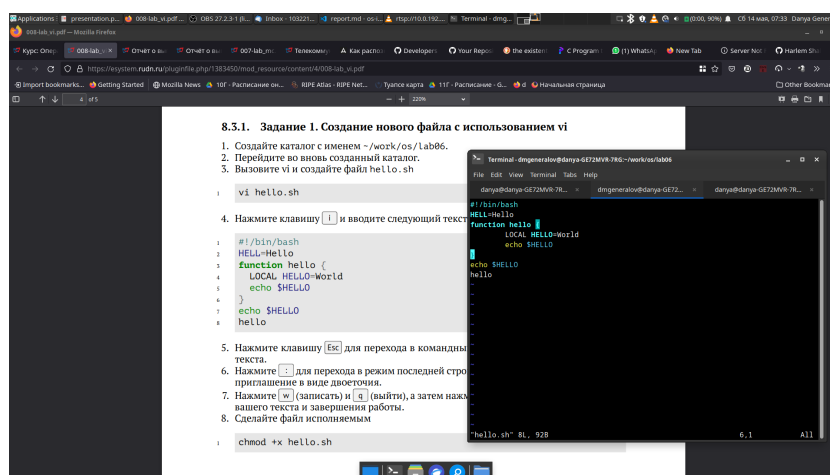


Рис. 4.1: Окно после ввода текста

После этого нужно закрыть редактор, сначала нажав клавишу `:`, чтобы перейти в режим ввода команд, а затем ввести `wq` для сохранения текста и выхода из редактора. После этого нужно сделать текстовый файл исполняемым, однако сейчас он не будет работать из-за синтаксических ошибок, как показано на рис. 4.2.

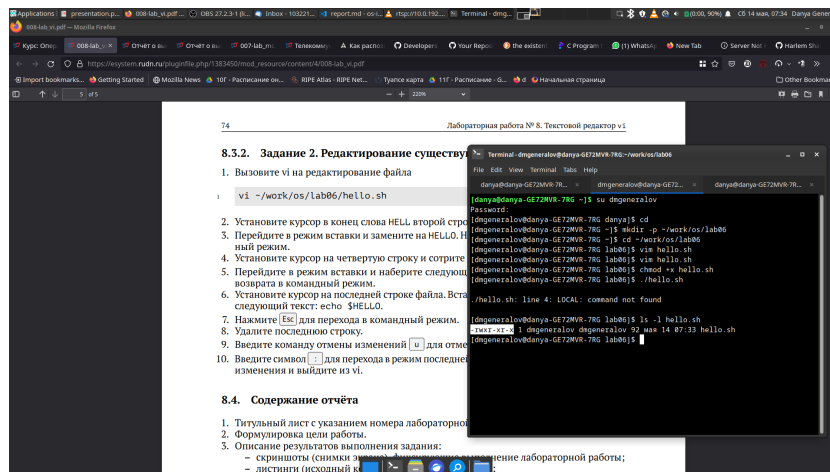


Рис. 4.2: Делаем файл исполняемым

Для того, чтобы исправить ошибки, нужно изменить некоторые строки. Для этого можно пользоваться режимом вставки, а также командами редактирования текста вроде `s/LOCAL/local`. Также можно опробовать команду удаления строки, `dd`, и команду отмены, `u`. Итоговое состояние текста видно на рис. 4.3.

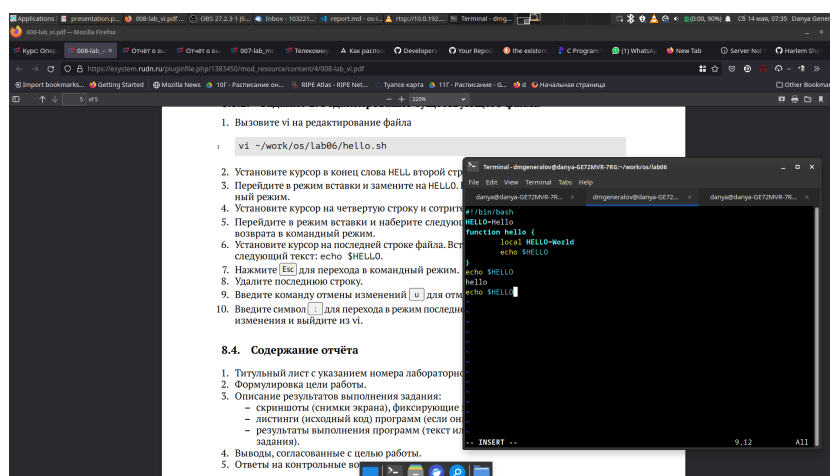


Рис. 4.3: Исправленный текст

После этого можно сохранить этот файл снова и попробовать его выполнить – на этот раз он запускается правильно, как на рис. 4.4.

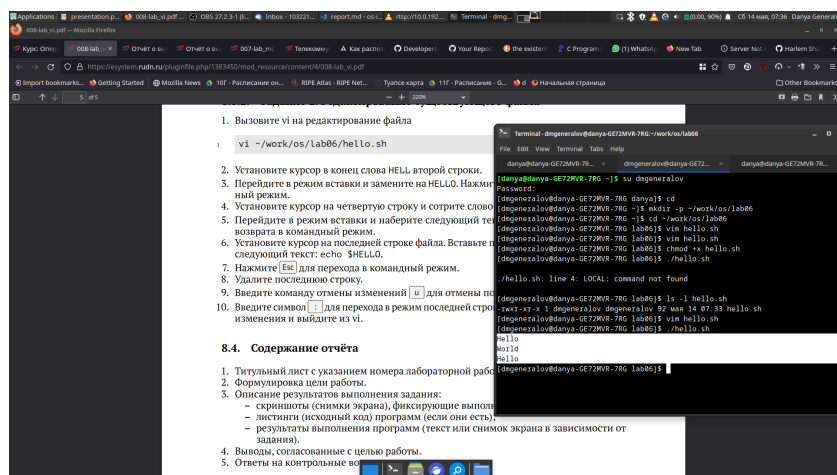


Рис. 4.4: Запускаем файл

5 Контрольные вопросы

1. Дайте краткую характеристику режимам работы редактора vi.

- Командный режим – режим по умолчанию, позволяющий просматривать файл, выполнять команды клавишами и переходить в другие режимы
- Режим ввода команд – позволяет вводить команды, описанные текстом, в том числе команды сохранения и выхода
- Режим вставки – позволяет вводить и удалять текст, написанный вручную, в открытый файл
- Визуальный режим – позволяет выделять кусок текста по отрезку, прямоугольнику или по строкам, и выполнять те же самые действия, как и в командном режиме.

2. Как выйти из редактора, не сохраняя произведённые изменения?

Для этого нужно зайти в режим ввода команд, нажав :, а затем ввести qa!, что закроет все окна и выйдет из редактора без сохранения изменений. Можно узнать это, если нажать Ctrl+C в командном режиме.

3. Назовите и дайте краткую характеристику командам позиционирования.

Команды позиционирования используются в командном режиме, чтобы перемещать курсор в файле. Для нас полезны следующие команды:

- 0 – перемещает курсор в начало строки
- \$ – перемещает курсор в конец строки
- G – перемещает курсор в конец файла

- gg – перемещает курсор в начало файла
- <n>G – перемещает курсор в конец строки с номером n, где начальная строка файла имеет номер 1

4. Что для редактора vi является словом?

В контексте команд W и B, которые переходят по словам влево и вправо, слово – эта любая последовательность знаков, разделенная пробелом, знаком табуляции или знаком переноса строки. Для команд w и b, спецсимволы воспринимаются как пробельные символы и не считаются частью слова.

5. Каким образом из любого места редактируемого файла перейти в начало (конец) файла?

Для этого можно использовать команду 1G, которая переходит на первую строку, и G, которая переходит на последнюю строку.

6. Назовите и дайте краткую характеристику основным группам команд редактирования.

Существуют следующие группы команд:

- вставка текста внутри строки: i, I a, A
- вставка текста на новую строку: o, O
- удаление текста: x, d
- передвижение в истории изменений: u, Ctrl+R
- работа с буфером обмена: y, Y, p, P
- замена текста: r, R, c
- поиск: /, ?

7. Необходимо заполнить строку символами \$. Каковы ваши действия?

Используя стандартную ширину строки в 80 колонок, можно набрать команду 80i, нажать клавишу \$, затем выйти из режима ввода через Esc, чтобы получить 80 символов \$ в строке.

8. Как отменить некорректное действие, связанное с процессом редактирования?

Чтобы отменить последнее изменение, можно использовать команду `u`. Чтобы отменить отмену изменения, можно использовать команду `Ctrl+r`.

9. Назовите и дайте характеристику основным группам команд режима последней строки.

В режиме ввода команд можно делать очень много различных действий. Можно даже делать собственные команды, выполняющие полезные наборы действий. Однако самые часто употребляемые команды – это:

- `w, wq, q!` – для записи в файл, записи и выхода из программы, или выхода без записи
- `set` – для изменения настроек программы
- `d, m, t` – для редактирования наборов строк по номерам
- `split, vsplit, term, close` – для управления окнами
- `help, help <topic>` – для получения помощи по командам или по теме

10. Как определить, не перемещая курсора, позицию, в которой заканчивается строка?

Иными словами, нужно определить, какая длина текущей строки. Для этого можно использовать команду `:.w !wc -с`. Эта команда передаст текущую строку на вход команде `wc -с`, которая посчитает количество символов во входном потоке.

11. Выполните анализ опций редактора `vi` (сколько их, как узнать их назначение и т.д.).

Согласно man-странице по `vi` на моей системе, программа принимает следующие 8 опций командной строки: `-с`, `-l`, `-r`, `-R`, `-s`, `-t`, `-V` и `-w`. Некоторые из

этих опций используются для того, чтобы использовать vi в неинтерактивном режиме, а некоторые устанавливают настройки, которые используются для изменения поведения vi в пользовательском режиме – чтобы определить, что именно делает какая опция, нужно посмотреть man-страницу.

12. Как определить режим работы редактора vi?

Текущий активный режим можно узнать по надписи внизу экрана – если нет текста, это командный режим, если есть один знак : – режим ввода команд, если есть слово INSERT – режим вставки, а если есть слово VISUAL – визуальный режим.

13. Постройте граф взаимосвязи режимов работы редактора vi.

Начало -> Командный режим

Командный режим -[клавиша i]> Режим вставки

Командный режим -[клавиша o]> Режим вставки

Режим вставки -[клавиша Esc]> Командный режим

Командный режим -[клавиша :]> Режим ввода команд

Режим ввода команд -[клавиша Enter]> Командный режим

Режим ввода команд -[клавиша Esc]> Командный режим

Командный режим -[клавиша :]> Режим ввода команд

Режим ввода команд -[q!]> Конец

Командный режим -[клавиша v]> Визуальный режим

Командный режим -[клавиша V]> Визуальный режим (строками)

Командный режим -[Ctrl+V]> Визуальный режим (прямоугольником)

Визуальный режим -[клавиша Esc]> Командный режим

Командный режим -[клавиша Esc]> Командный режим

6 Выводы

Мы попробовали воспользоваться текстовым редактором VI, который присутствует на многих системах и поэтому полезен для редактирования текста вроде конфигурационных файлов. Используя этот редактор, можно писать текст, а также редактировать его с помощью широкого набора команд. Из-за этого широкого набора команд, этот редактор сложно использовать в полную силу, но теперь мы имеем достаточное представление о нем, чтобы иметь возможность использовать его на базовом уровне.