

Лабораторная работа №8

Простейший вариант выполнения лабораторной работы

Атанесов Александр Николаевич

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	20
	Список литературы	23

Список иллюстраций

4.1	Использую Браузер	8
4.2	Использую Браузер	9
4.3	Использую команду mkdir	10
4.4	Использую команду cd	10
4.5	Использую команду vi	10
4.6	Использую клавишу “I”	11
4.7	Использую Буфер обмена	12
4.8	Использую команду Shift+:+W+Q	13
4.9	Использую команду chmod +x	13
4.10	Использую команду vi	14
4.11	Использую команду INSERT	14
4.12	Использую команду INSERT	15
4.13	Использую команду INSERT	16
4.14	Использую команду INSERT	17
4.15	Использую команду U	18
4.16	Использую команду ESC+SHIFT+:+W+Q	19

Список таблиц

3.1	Описание некоторых каталогов файловой системы GNU Linux . .	7
-----	---	---

1 Цель работы

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

2 Задание

1. Взаимодействовать с редактором vi;

3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. [3.1] приведено краткое описание стандартных каталогов Unix.

Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя

Более подробно об Unix см. в [1–6].

4 Выполнение лабораторной работы

1. Изучаю теоретический материал. (рис. [4.1])

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 <имя_файла>

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.

Рис. 4.1: Используя Браузер

4.1

2. Изучаю материал о редакторе vi. (рис. [4.2])

8.2.1.1. Команды управления курсором





Команды управления курсором приведены в табл. 8.1.

Таблица 8.1





Команды управления курсором

Курсор влево	Курсор вправо	Курсор вверх	Курсор вниз
 (клавиша Backspace)	 (клавиша «пробел»)		
 	 	 	 

8.2.1.2. Команды позиционирования

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

8.2.1.3. Команды перемещения по файлу

-  — перейти на пол-экрана вперёд;
-  — перейти на пол-экрана назад;
-  — перейти на страницу вперёд;
-  — перейти на страницу назад.

8.2.1.4. Команды перемещения по словам¹









-  или  — перейти на слово вперёд;
- n  или n  — перейти на n слов вперёд;
-  или  — перейти на слово назад;
- n  или n  — перейти на n слов назад.

Рис. 4.2: Используя Браузер

4.2

3. Создаю каталог os/lab06. (рис. [4.3])

```
[aatanesov@fedora work]$ mkdir os/lab06
```

Рис. 4.3: Использую команду mkdir

4.3

4. Открываю lab06. (рис. [4.4])

```
[aatanesov@fedora os]$ cd lab06
```

Рис. 4.4: Использую команду cd

4.4

5. Создаю и открываю файл через редактор vi. (рис. [4.5])

```
[aatanesov@fedora lab06]$ vi hello.sh
```

Рис. 4.5: Использую команду vi

4.5

6. Использую функцию для вставки текста. (рис. [4.6])

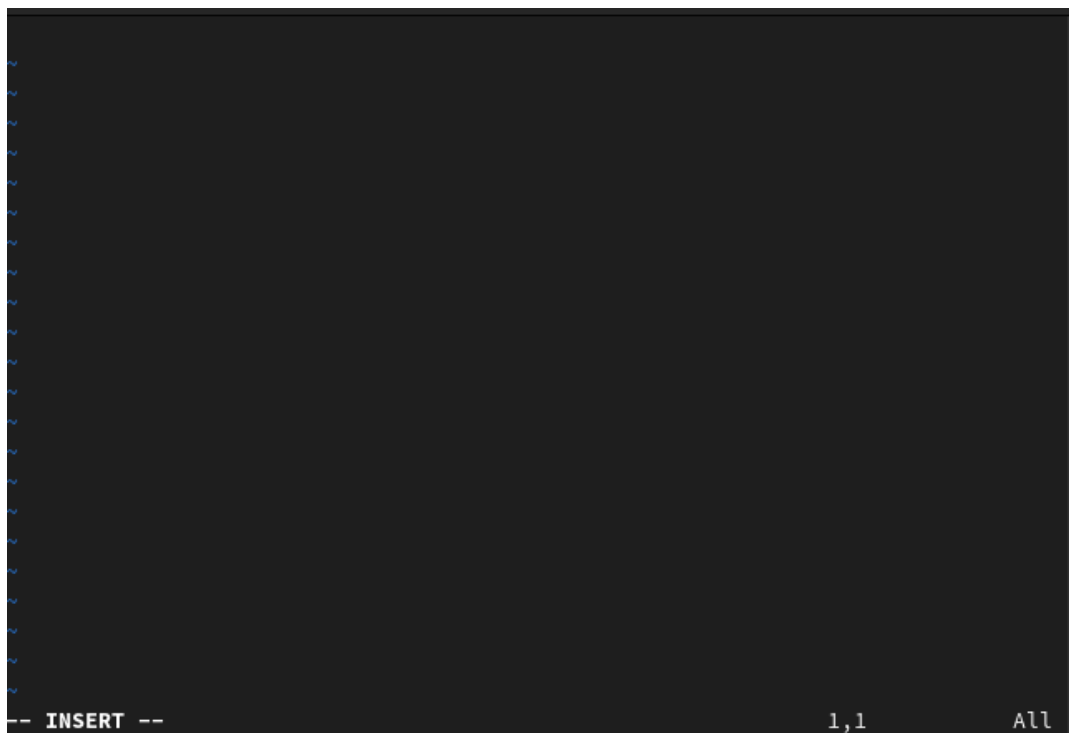
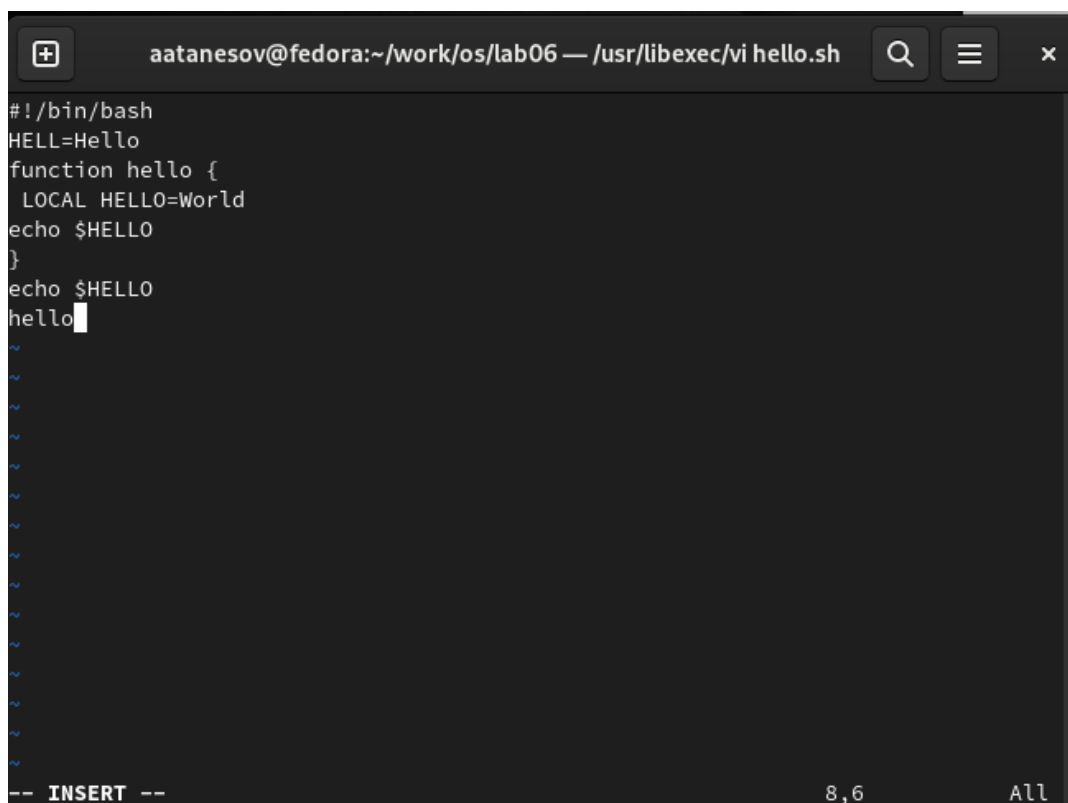


Рис. 4.6: Использу клавишу "I"

4.6

7. Вставляю текст, данный в лабораторной работе . (рис. [4.7])



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

Рис. 4.7: Используя Буфер обмена

4.7

8. Выхожу и сохраняю файл . (рис. [4.8])


```
[aatanesov@fedora lab06]$ vi hello.sh
```

Рис. 4.10: Используя команду vi

4.10

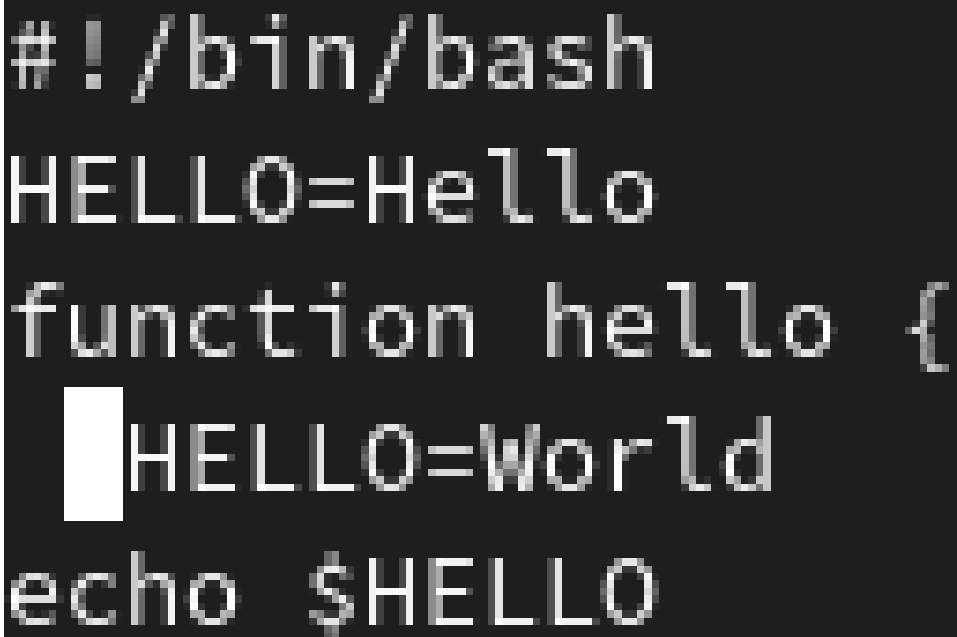
11. Меняю HELL на HELLO. (рис. [4.11])

```
#!/bin/bash  
HELLO=Hello  
function hello {
```

Рис. 4.11: Используя команду INSERT

4.11

12. Стираю слово LOCAL. (рис. [4.12])

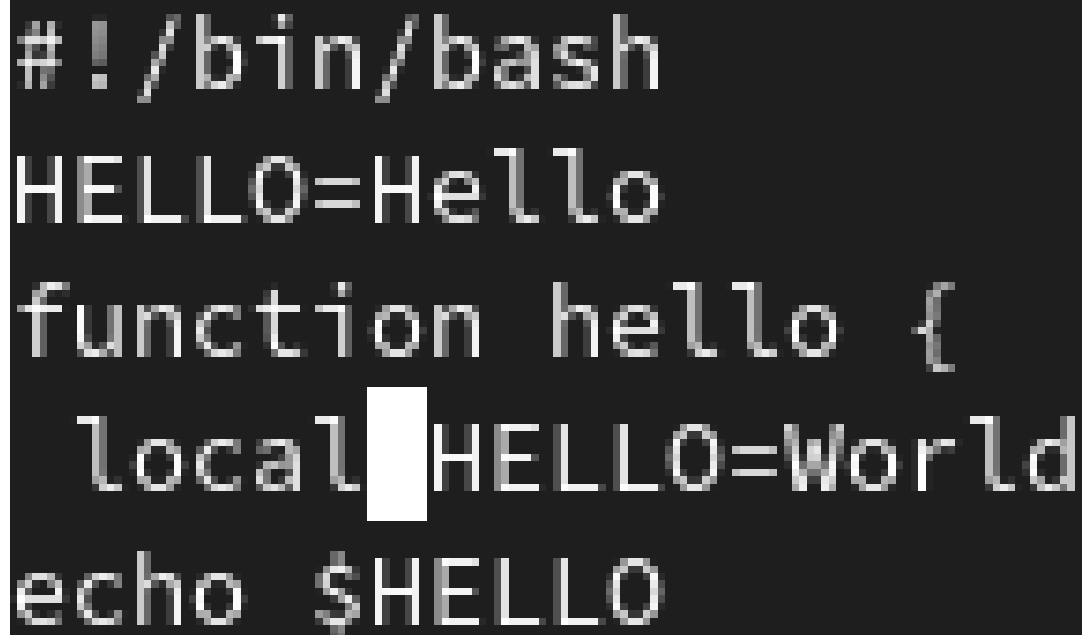


```
#!/bin/bash
HELLO=Hello
function hello {
  HELLO=World
  echo $HELLO
}
```

Рис. 4.12: Используя команду INSERT

4.12

13. Пишу вместо него local. (рис. [4.13])



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

Рис. 4.13: Используя команду INSERT

4.13

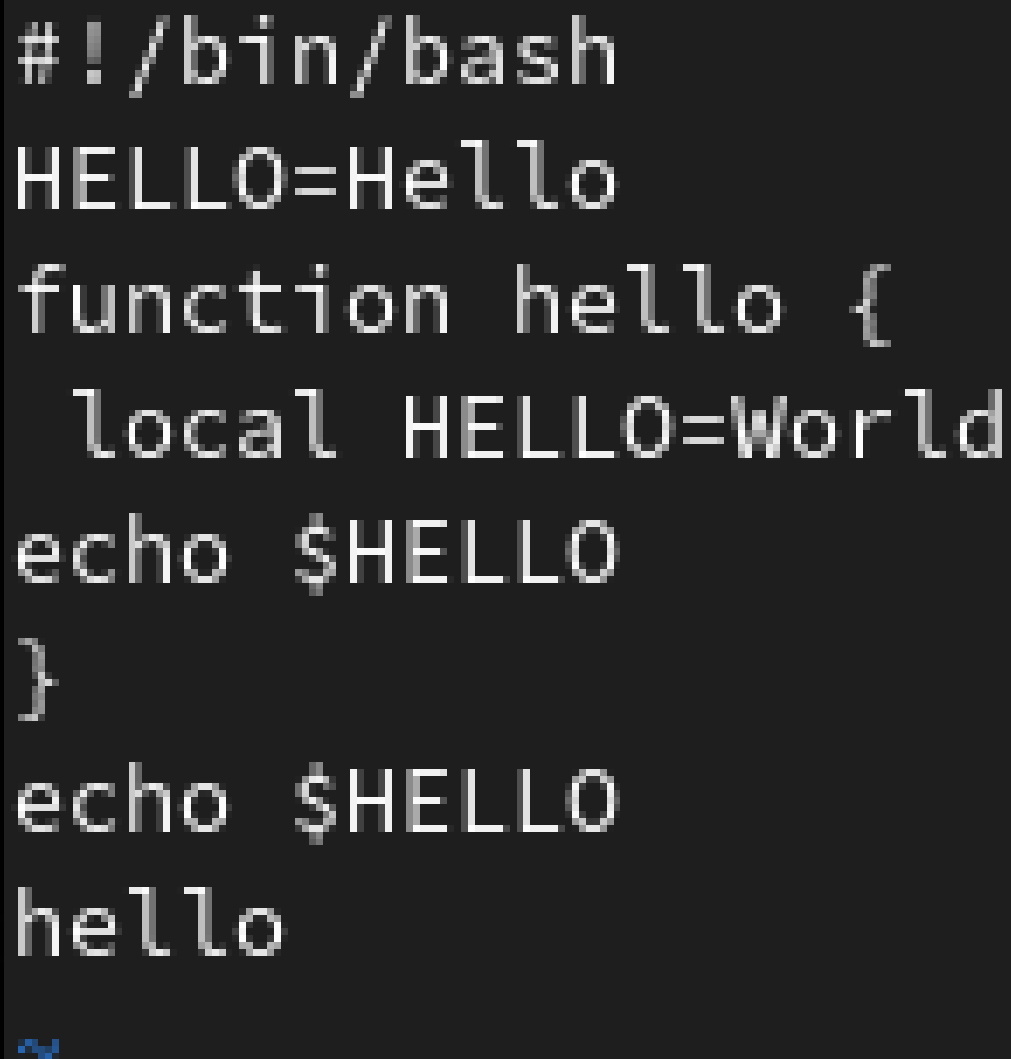
14. Перехожу в конец файла и пишу echo \$HELLO. (рис. [4.14])


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

Рис. 4.14: Используя команду INSERT

4.14

15. Отменяю последнюю команду. (рис. [4.15])

A terminal window with a dark background and light gray text. The text shows a shell script being executed. The script starts with a shebang line, sets a variable, defines a function, and then calls the function. The output of the function is displayed. A blue cursor is visible at the end of the last line.

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

Рис. 4.15: Используя команду U

4.15

16. Сохраняю и выхожу из файла hello.sh. (рис. [4.16])

5 Выводы

- Узнал о редакторе vi и научился пользоваться его командами. # Ответы на контрольные вопросы
- 1. Режимы работы редактора vi:
 - Командный режим (Command mode) - используется для перемещения по тексту, удаления, копирования и вставки текста, поиска и замены.
 - Режим вставки (Insert mode) - используется для ввода текста.
 - Режим замены (Replace mode) - используется для замены символов в тексте.
 - Режим визуализации (Visual mode) - используется для выделения текста.
- 2. Для выхода из редактора без сохранения изменений нужно нажать комбинацию клавиш Esc :q! и нажать Enter.
- 3. Команды позиционирования:
 - h - перемещение курсора влево на один символ.
 - j - перемещение курсора вниз на одну строку.
 - k - перемещение курсора вверх на одну строку.
 - l - перемещение курсора вправо на один символ.
 - 0 - перемещение курсора в начало строки.
 - \$ - перемещение курсора в конец строки.

- w - перемещение курсора на начало следующего слова.
- b - перемещение курсора на начало предыдущего слова.
- 4. В редакторе vi словом является последовательность символов, разделенная пробелами или символами перевода строки.
- 5. Для перехода в начало файла нужно нажать комбинацию клавиш Esc gg, для перехода в конец файла - Esc G.
- 6. Основные группы команд редактирования:
 - Удаление (Deletion) - команды для удаления символов, слов, строк.
 - Копирование и вставка (Copy and Paste) - команды для копирования и вставки текста.
 - Замена (Replace) - команды для замены символов.
 - Поиск и замена (Search and Replace) - команды для поиска и замены текста.
 - Изменение (Change) - команды для изменения текста.
 - Выделение (Selection) - команды для выделения текста.
- 7. Для заполнения строки символами \$ нужно находиться в командном режиме, поставить курсор на начало строки и ввести команду 80i\$ (если нужно заполнить 80 символов).
- 8. Для отмены некорректного действия нужно нажать комбинацию клавиш Esc u.
- 9. Основные группы команд режима последней строки:
 - Сохранение (Saving) - команда для сохранения изменений в файле.
 - Выход (Exit) - команда для выхода из редактора.

- Поиск и замена (Search and Replace) - команды для поиска и замены текста.
- Открытие файла (Open file) - команда для открытия нового файла.
- 10. Для определения позиции конца строки без перемещения курсора нужно нажать комбинацию клавиш \$.
- 11. Редактор vi имеет множество опций, которые можно установить при запуске редактора или в процессе работы. Опции можно узнать, введя команду :set в командном режиме.
- 12. Режим работы редактора vi можно определить по символу, который отображается в левом нижнем углу экрана:
 - – INSERT – - режим вставки.
 - – REPLACE – - режим замены.
 - – VISUAL – - режим визуализации.
- Пустая строка - командный режим.
- 13. Граф взаимосвязи режимов работы редактора vi: [Командный режим] -> [Режим вставки] [Командный режим] -> [Режим замены] [Командный режим] -> [Режим визуализации] [Режим вставки] -> [Командный режим] [Режим замены] -> [Командный режим] [Режим визуализации] -> [Командный режим]

Список литературы

1. GNU Bash Manual [Электронный ресурс]. Free Software Foundation, 2016.
URL: <https://www.gnu.org/software/bash/manual/>.
2. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
5. Таненбаум Э. Архитектура компьютера. 6-е изд. СПб.: Питер, 2013. 874 с.
6. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.