

Отчёт по лабораторной работе №9

Дисциплина: Операционные системы

Шмырин Михаил Сергеевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	16
4	Ответы на контрольные вопросы	17

Список иллюстраций

2.1	Emacs	6
2.2	Emacs	6
2.3	Создание файла lab07	7
2.4	Ввод текста	7
2.5	Вырезаем целую строку	8
2.6	Вставляем строку в конец файла	8
2.7	Выделяем область текста	9
2.8	Вставляем область в конец файла	9
2.9	Вырезаем область	10
2.10	Отмена последнего действия	10
2.11	Вывод списка активных буферов	11
2.12	Переключение на другой буфер	11
2.13	Закрываю окно со списком буферов	12
2.14	Переключение на другой буфер	12
2.15	Деление фрейма на 4 части	12
2.16	Ввод текста	13
2.17	Режим поиска	14
2.18	Результат поиска	14
2.19	Замена	15

Список таблиц

1 Цель работы

Познакомиться с операционной системой Linux.Получить практические навыки работы с редактором Emacs.

2 Выполнение лабораторной работы

1. Открываю emacs.(рис. 2.1) (рис. 2.2)

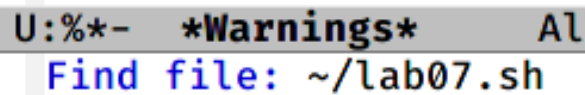
```
midhs@msshmihrin:~$ emacs &  
[1] 4489  
midhs@msshmihrin:~$ emacs &  
[2] 4667  
[1]   Завершён      emacs
```

Рис. 2.1: Emacs



Рис. 2.2: Emacs

2. Создаю файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (рис. 2.3)



```
U:%*- *Warnings* Al
Find file: ~/lab07.sh
```

Рис. 2.3: Создание файла lab07

3. Набираю текст: (рис. 2.4)

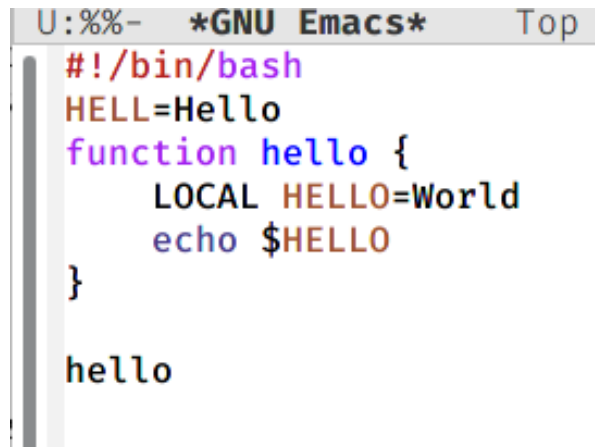


```
U:%%- *GNU Emacs* Top L5
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello

U:**- lab07.sh All L8
```

Рис. 2.4: Ввод текста

4. Сохраняю файл с помощью комбинации Ctrl-x Ctrl-s.
5. Прodelываю с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш.
- 5.1. Вырезаю одной командой целую строку (C-k).(рис. 2.5)



```
U:%%- *GNU Emacs* Top
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}

hello
```

Рис. 2.5: Вырезаем целую строку

5.2. Вставляю эту строку в конец файла (C-y).(рис. 2.6)



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

hello
echo $HELLO
```

Рис. 2.6: Вставляем строку в конец файла

5.3. Выделяю область текста (C-space).(рис. 2.7)


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

Рис. 2.7: Выделяем область текста

5.4. Копирую область в буфер обмена (M-w).

5.5. Вставляю область в конец файла. (рис. 2.8)

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

hello
echo $HELLO
function hello {
    LOCAL HELLO=World
```

Рис. 2.8: Вставляем область в конец файла

5.6. Вновь выделяю эту область и на этот раз вырезаю её (C-w).2.9)

```
#!/bin/bash
HELL=Hello

    echo $HELLO
}

hello
echo $HELLO
function hello {
    LOCAL HELLO=World
    █
```

Рис. 2.9: Вырезаем область

5.7. Отменяю последнее действие (C-/).(рис. 2.10)

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

hello
echo $HELLO
    function hello {
        LOCAL HELLO=World
```

Рис. 2.10: Отмена последнего действия

6. Учусь использовать команды по перемещению курсора.

6.1. Перемещаю курсор в начало строки (C-a).

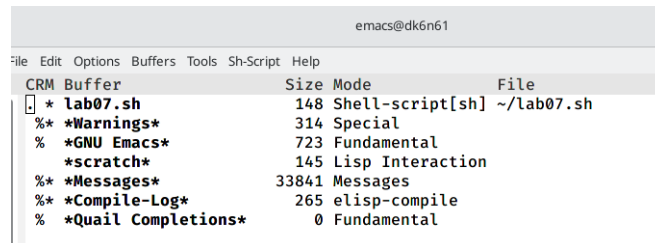
6.2. Перемещаю курсор в конец строки (C-e).

6.3. Перемещаю курсор в начало буфера (M-<).

6.4. Перемещаю курсор в конец буфера (M->).

7. Управление буферами.

7.1. Вывожу список активных буферов на экран (C-x C-b) (рис. 2.11)

A screenshot of the Emacs buffer list window. The title bar says 'emacs@dk6n61'. The menu bar includes 'File Edit Options Buffers Tools Sh-Script Help'. The buffer list shows several buffers: 'lab07.sh' (148 characters, Shell-script[sh], ~/lab07.sh), '*Warnings*' (314 characters, Special), '*GNU Emacs*' (723 characters, Fundamental), '*scratch*' (145 characters, Lisp Interaction), '*Messages*' (33841 characters, Messages), '*Compile-Log*' (265 characters, elisp-compile), and '*Quail Completions*' (0 characters, Fundamental).

CRM	Buffer	Size	Mode	File
	* lab07.sh	148	Shell-script[sh]	~/lab07.sh
%*	*Warnings*	314	Special	
%	*GNU Emacs*	723	Fundamental	
%	*scratch*	145	Lisp Interaction	
%*	*Messages*	33841	Messages	
%*	*Compile-Log*	265	elisp-compile	
%	*Quail Completions*	0	Fundamental	

Рис. 2.11: Вывод списка активных буферов

7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых буферови переключитесь на другой буфер.(рис. 2.12)

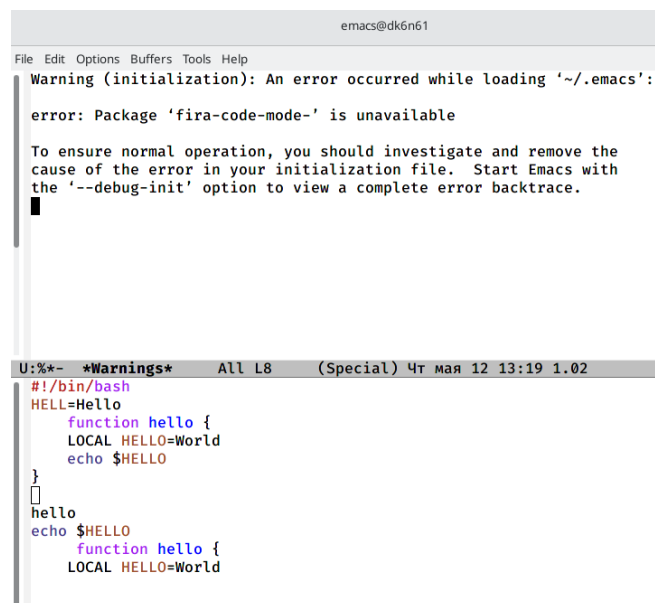
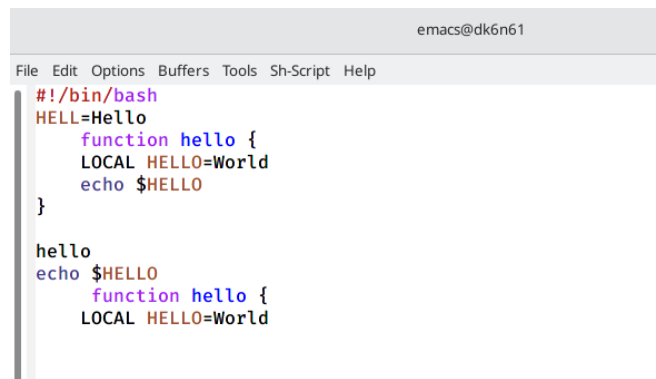
A screenshot of the Emacs buffer switch window. The title bar says 'emacs@dk6n61'. The menu bar includes 'File Edit Options Buffers Tools Help'. The main text area shows a warning message: 'Warning (initialization): An error occurred while loading '~/.emacs': error: Package 'fira-code-mode-' is unavailable'. Below the warning, there is a message: 'To ensure normal operation, you should investigate and remove the cause of the error in your initialization file. Start Emacs with the '--debug-init' option to view a complete error backtrace.' At the bottom, there is a buffer list showing 'U:%*- *Warnings*' (All L8, (Special), 4т мая 12 13:19 1.02). The buffer list also shows a list of buffers: '#!/bin/bash', 'HELL=Hello', 'function hello {', 'LOCAL HELLO=World', 'echo \$HELLO', '}', 'hello', 'echo \$HELLO', 'function hello {', 'LOCAL HELLO=World'.

Рис. 2.12: Переключение на другой буфер

7.3. Закрываю это окно (C-x 0).(рис. 2.13)

A screenshot of the Emacs editor window titled 'emacs@dk6n61'. The menu bar includes 'File', 'Edit', 'Options', 'Buffers', 'Tools', 'Sh-Script', and 'Help'. The buffer content is a shell script:

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

hello
echo $HELLO
function hello {
  LOCAL HELLO=World
```

Рис. 2.13: Закрываю окно со списком буферов

7.4. Теперь вновь переключаюсь между буферами, но уже без вывода их списка наэкран (C-x b).(рис. 2.14)

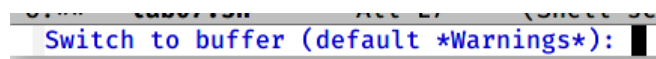
A screenshot of the Emacs editor showing the command 'Switch to buffer (default *Warnings*):' in the command line.

Рис. 2.14: Переключение на другой буфер

8. Управление окнами.

8.1. Делю фрейм на 4 части: разделите фрейм на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по горизонтали (C-x 2)(рис. 2.15)

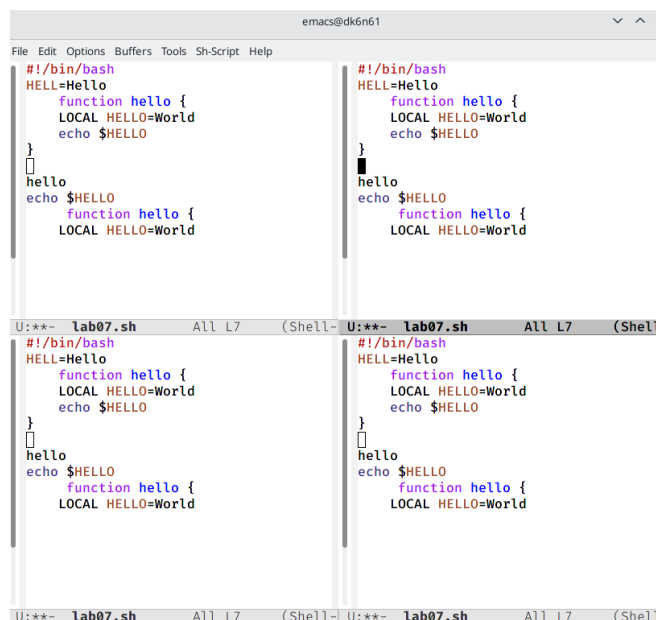
A screenshot of the Emacs editor showing a 2x2 window layout. Each of the four windows contains the same shell script as in Figure 2.13. The status bar at the bottom of each window shows 'U:*** lab07.sh All L7 (Shell-)'. The Emacs title bar at the top shows 'emacs@dk6n61'.

Рис. 2.15: Деление фрейма на 4 части

8.2. В каждом из четырёх созданных окон открываю новый буфер (файл) и ввожу текст.(рис. 2.16)

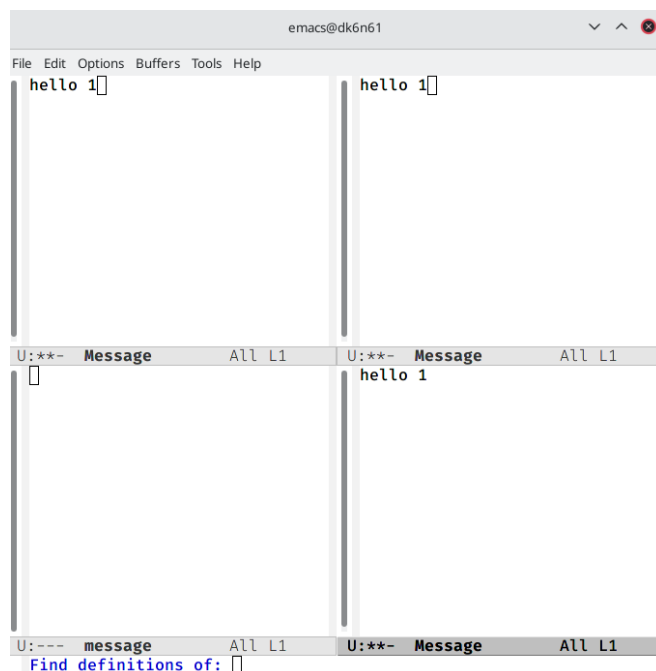


Рис. 2.16: Ввод текста

9. Режим поиска

9.1.Переключаюсь в режим поиска (C-s) и нахожу несколько слов, присутствующих в тексте.(рис. 2.17)

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

hello
echo $HELLO
function hello {
    LOCAL HELLO=World
```

Рис. 2.17: Режим поиска

9.2. Переключаюсь между результатами поиска, нажимая C-s.

9.3. Выхожу из режима поиска, нажав C-g.

9.4. Перехожу в режим поиска и замены (M-%), ввожу текст, который следует найти и заменить, нажимаю Enter, затем ввожу текст для замены. После того как будут подсвечены результаты поиска(рис. 2.18), нажимаю ! для подтверждения замены.(рис. 2.19)

```
hello
echo $HELLO
function hello {
    LOCAL HELLO=World
```

Рис. 2.18: Результат поиска

```
piece
echo $PIECE
function piece {
    LOCAL PIECE=World
}
```

Рис. 2.19: Замена


3 Выводы

Я познакомился с операционной системой Linux, получил практические навыки работы с редактором Emacs.

4 Ответы на контрольные вопросы

1. Emacs – один из наиболее мощных и широко распространенных редакторов, используемых в мире Unix. По популярности он соперничает с редактором vi и его клонами. В зависимости от ситуации, Emacs может быть текстовым редактором; программой для чтения почты и новостей Usenet; интегрированной средой разработки (IDE); операционной системой и т.д. Все это разнообразие достигается благодаря архитектуре Emacs, которая позволяет расширять возможности редактора при помощи языка Emacs Lisp. На языке C написаны лишь самые базовые и низкоуровневые части Emacs, включая полнофункциональный интерпретатор языка Lisp. Таким образом, Emacs имеет встроенный язык программирования, который может использоваться для настройки, расширения и изменения поведения редактора. В действительности, большая часть того редактора, с которым пользователи Emacs работают в наши дни, написана на языке Lisp.
2. Основную трудность для новичков при освоении данного редактора могут составлять большое количество команд, комбинаций клавиш, которые не получится все запомнить с первого раза и поэтому придется часто обращаться к справочным материалам.
3. Буфер – это объект, представляющий собой текст. Если имеется несколько буферов, то редактировать можно только один. Обычно буфер считывает данные из файла или записывает в файл данные из буфера. Окно – это область экрана, отображающая буфер. При запуске редактора отображается одно окно, но при обращении к некоторым функциям могут открыться

дополнительные окна. Окна Emacs окна графической среды XWindow – разные вещи. Одно окно XWindow может быть разбито на несколько окон в смысле Emacs, в каждом из которых отображается отдельный буфер.

4. Да, можно.
5. При запуске Emacs по умолчанию создаются следующие буферы: «scratch» (буфер для несохраненного текста) «Messages» (журнал ошибок, включающий также информацию, которая появляется в области EchoArea) «GNU Emacs» (справочный буфер о редакторе).
6. C-c | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и нажимаю «|» C-c C-| сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и, удерживая «ctrl», нажимаю «|».
7. Чтобы поделить окно на две части необходимо воспользоваться комбинацией «Ctrl-x 3» (по вертикали) или «Ctrl-x 2» (по горизонтали).
8. Настройки Emacs хранятся в файле emacs.
9. По умолчанию клавиша «» удаляет символ перед курсором, но в редакторе emacs можно переназначить. Для этого необходимо изменить конфигурацию файла emacs.
10. Более удобным я считаю редактор emacs, потому что в нем проще открывать другие файлы, можно использовать сразу несколько окон, нет «Командного режима», «Режима ввода», «Режима командной строки», которые являются немного непривычными и в какой-то степени неудобными.