

# Лабораторная работа №9

## Отчёт по лабораторной работе №9

Макарова Анастасия Михайловна

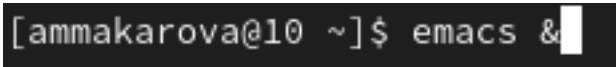
### Содержание

### Цель работы

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

### Выполнение лабораторной работы


1. Откроем редактор emacs помощью команды emacs \$ (Рис.1).



```
[ammakarova@10 ~]$ emacs &
```

Рис.1

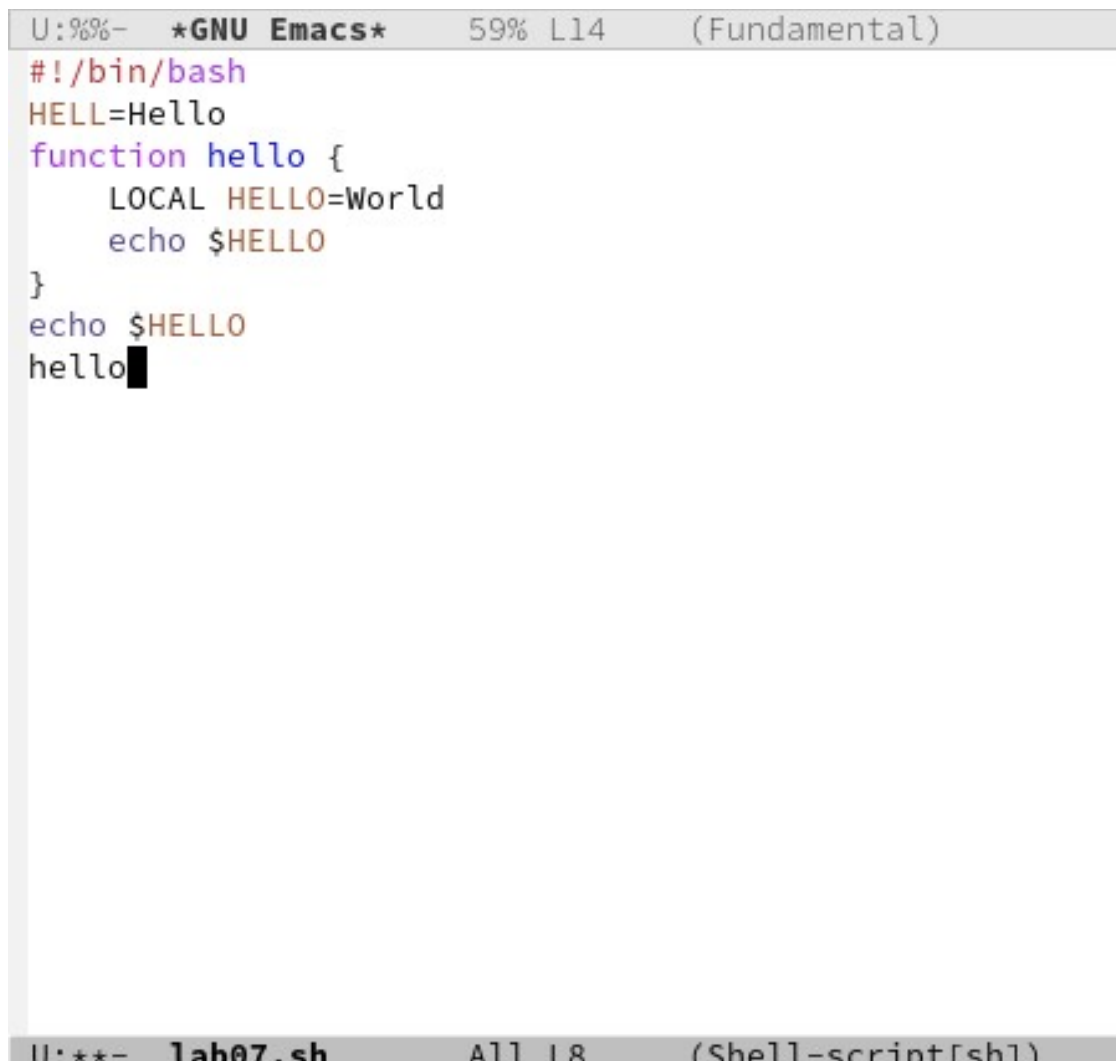
2. Создаем файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (Рис.2).



```
Find file: ~/lab07.sh
```

Рис.2

3. Наберем предложенный нам текст (Рис.3).

A screenshot of the GNU Emacs editor interface. The top status bar shows 'U:%%- \*GNU Emacs\* 59% L14 (Fundamental)'. The main window displays a shell script with the following content: 

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

 The bottom status bar shows 'U:\*+- lab07.sh All 18 (Shell-script[sh])'.

```
U:%%- *GNU Emacs* 59% L14 (Fundamental)
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
U:*+- lab07.sh All 18 (Shell-script[sh])
```

Рис.3

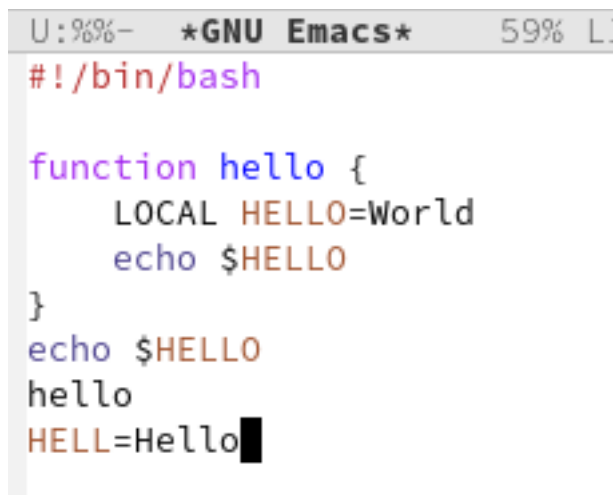
4. Сохраним файл с помощью комбинации Ctrl-x Ctrl-s.
5. Прделаем с текстом стандартные процедуры редактирования:
  - 5.1. Вырежем одной командой целую строку с помощью комбинации клавиш Ctrl-k (Рис.4).



```
U:%%- *GNU Emacs* 59% L14
#!/bin/bash
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
```

Рис.4

5.2. Вставим эту строку в конец файла с помощью комбинации клавиш Ctrl-y (Рис.5).



```
U:%%- *GNU Emacs* 59% L14
#!/bin/bash
function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
```

Рис.5

5.3. Выделим область текста с помощью комбинации клавиш Ctrl-space (Рис.6)

```
U:%%- *GNU Emacs* 59% L14
#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello

U:*- lab07.sh All L7
Mark set
```

Рис.6

5.4. Скопируем область в буфер обмена с помощью комбинации клавиш Alt-w.

5.5. Вставим область в конец файла с помощью комбинации клавиш Ctrl-y (Рис.7)

```
#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO
```

Рис.7

5.6. Вновь выделим эту область и на этот раз вырежем её с помощью комбинации клавиш Ctrl-w (Рис.8)

```
#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
```

Рис.8

5.7. Отменим последнее действие с помощью комбинации клавиш Ctrl-/ (Рис.9)

```
#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO
```

Рис.9

6. Научимся использовать команды по перемещению курсора:

6.1. Переместим курсор в начало строки с помощью комбинации клавиш Ctrl-a (Рис.10).

```
echo $HELLO
hello
HELL=Hello
echo $HELLO
```

Рис.10

6.2. Переместим курсор в конец строки с помощью комбинации клавиш Ctrl-e (Рис.11).

```
echo $HELLO
hello
HELL=Hello
echo $HELLO
```

Рис.11

6.3. Переместим курсор в начало буфера с помощью комбинации клавиш Alt-< (Рис.12).

```
#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO
```

Рис.12

6.4. Переместим курсор в конец буфера с помощью комбинации клавиш Alt-> (Рис.13).

```
#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO
```

Рис.13

## 7. Управляем буферами:

7.1. Выведем список активных буферов на экран с помощью комбинации клавиш Ctrl-x Ctrl-b (Рис.14).

-:*** lab07.sh		All L10	(Shell-script[bash])
CRM Buffer		Size	Mode File
[.] * lab07.sh		110	Shell-script[... ~/lab07.sh
% *GNU Emacs*		904	Fundamental
*scratch*		145	Lisp Interaction
%* *Messages*		1422	Messages

Рис.14

7.2. Переместимся во вновь открытое окно с помощью комбинации клавиш Ctrl-x со списком открытых буферов и переключимся на другой буфер (Рис.15).

```
U:--- *scratch* All L4 (Lisp Interaction ElDoc)
U -- utf-8-dos (alias: mule-utf-8-dos cp65001-dos)

UTF-8 (no signature (BOM))
Type: utf-8 (UTF-8: Emacs internal multibyte form)
EOL type: CR LF
This coding system encodes the following charsets:
  unicode
```

Рис.15

7.3. Закроем это окно с помощью комбинации клавиш Ctrl-x 0 (Рис.16).

```
#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO
```

Рис.16



7.4. Теперь вновь переключимся между буферами, но уже без вывода их списка на экран с помощью комбинации клавиш Ctrl-x b (Рис.17).

```
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO
```

-:\*\*\*- lab07.sh Bot L10 (Shell-script[bash])  
Switch to buffer (default \*Buffer List\*):

Рис.17

## 8. Управляем окнами:

8.1. Поделим фрейм на 4 части: разделим фрейм на два окна по вертикали с помощью клавиш Ctrl-x 3, а затем каждое из этих окон на две части по горизонтали с помощью комбинации клавиш Ctrl-x 2 (Рис.18, 19).

```

File Edit Options Buffers Tools Sh-Script Help
[Icons: New, Open, Save, Close, Undo, Cut, Copy, Paste, Find]

Welcome to GNU Emacs, one component of the GNU/Linux operating system.
To follow a link, click Mouse-1 on it, or move to it and type RET.
To quit a partially entered command, type Control-g.

Important Help menu items:
Emacs Tutorial      Learn basic Emacs keystroke commands
Read the Emacs Manual  View the Emacs manual using Info
\(Non\)Warranty      GNU Emacs comes with ABSOLUTELY NO WARRANTY
U:%%-  *GNU Emacs*  Top L6  (Fundamental)

echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO

#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO

-:***-  lab07.sh  Bot L10  (Shell-s -:***-  lab07.sh  All L10  (Shell
Switch to buffer (default *Buffer List*):

```

Puc.18

```

}
echo $HELLO
hello
HELL=Hello
echo $HELLO

#!/bin/bash

function hello {
    LOCAL HELLO=World
    echo $HELLO
}
echo $HELLO
hello
HELL=Hello
echo $HELLO

-:---  lab07.sh  Bot L10  (Shell-s -:---  lab07.sh  Top L8  (Shell
}
echo $HELLO
hello
HELL=Hello
echo $HELLO

#!/bin/bash

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

-:---  lab07.sh  Bot L10  (Shell-s -:---  lab07.sh  Top L8  (Shell

```

Puc.19

8.2. В каждом из четырёх созданных окон откроем новый буфер (файл) и введем несколько строк текста (Рис.20).

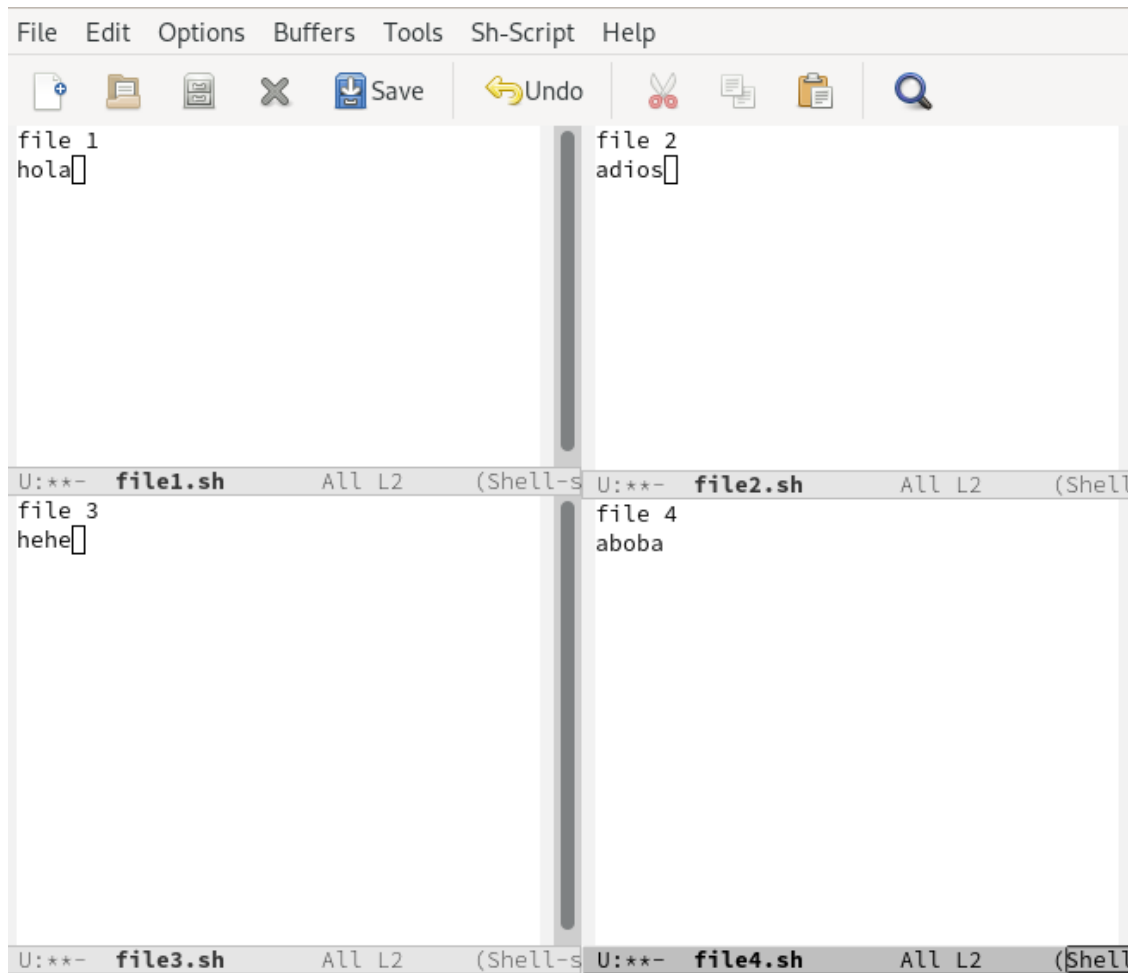


Рис.20

## 9. Управляем режимом поиска:

9.1. Переключимся в режим поиска с помощью клавиш Ctrl-s и найдем несколько слов, присутствующих в тексте (Рис.21).



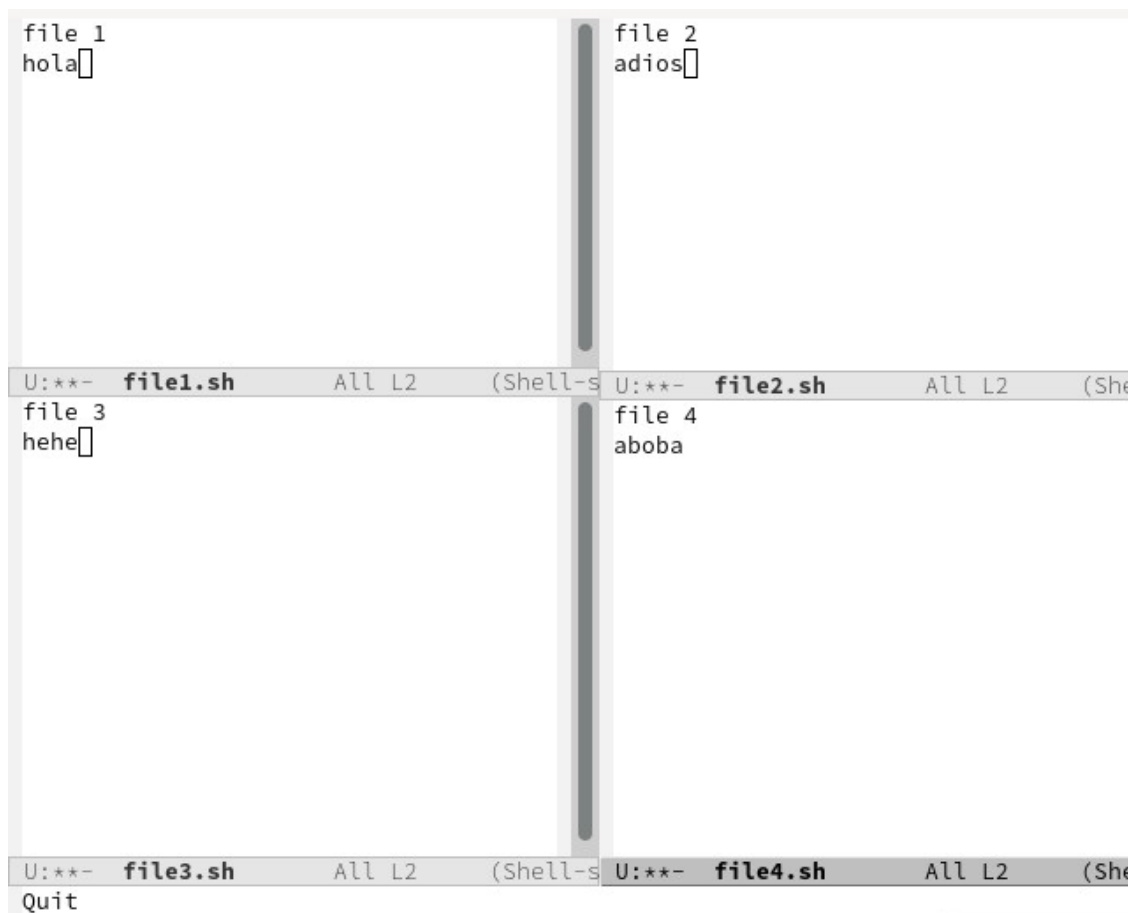
Рис.21

9.2. Переключаемся между результатами поиска, нажимая клавиши Ctrl-s (Рис.22).



Рис.22

9.3. Выйдем из режима поиска, нажав клавиши Ctrl-g (Рис.23).



*Рис.23*

9.4. Перейдем в режим поиска и замены с помощью клавиш Alt-%, введем текст, который следует найти и заменить, нажмем Enter, затем введем текст для замены. После того как будут подсвечены результаты поиска, нажмем ! для подтверждения замены (Рис.24, 25, 26).

U:\*\*~ **file1.sh** All L2 (Sh

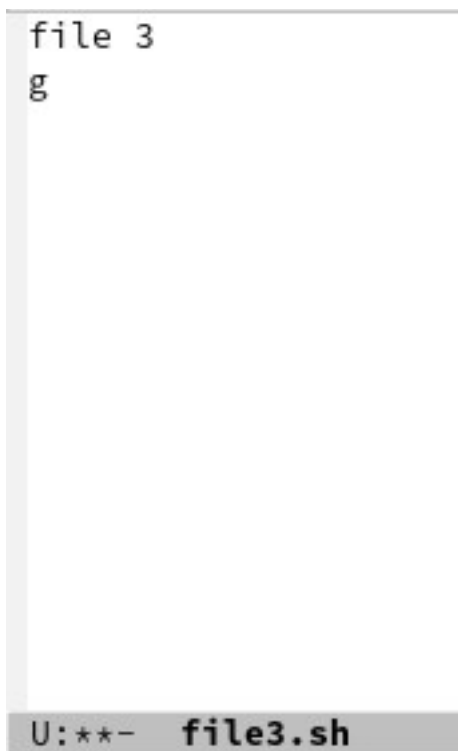
file 3

hehe

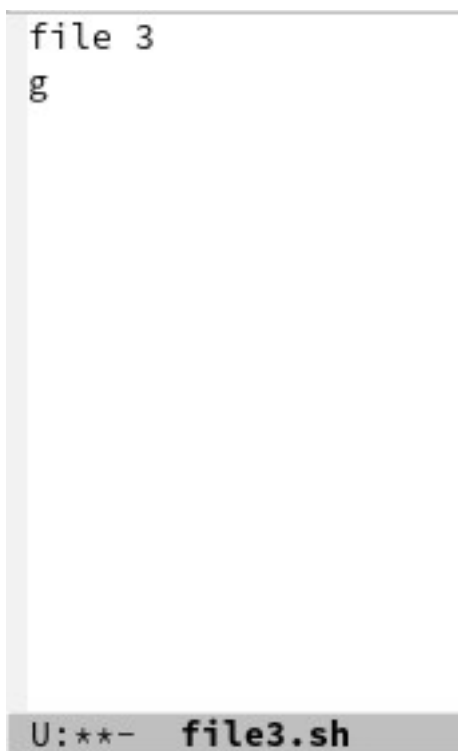
U:\*\*~ **file3.sh** All L2 (Sh

Query replace hehe with:

*Puc.24*



*Рис.25*



*Рис.25*

9.5. Испробуем другой режим поиска, нажав клавиши Alt-s o (Рис.27).



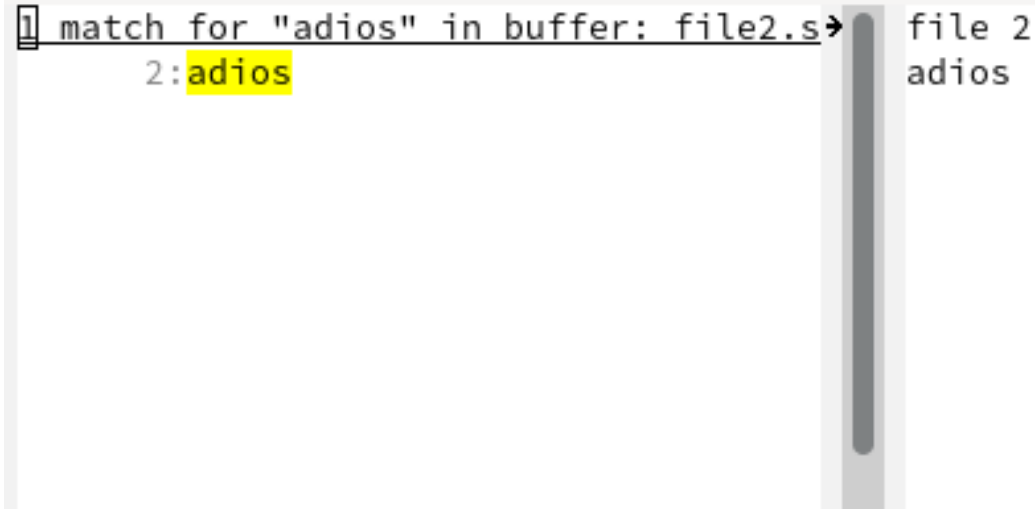


Рис.27

## Вывод

В ходе выполнения данной лабораторной работы я ознакомилась с операционной системой Linux и получила практические навыки работы с редактором Emacs.

## Контрольные вопросы

1. Кратко охарактеризуйте редактор emacs. Emacs – один из наиболее мощных и широко распространённых 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. Всё 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. Какие особенности данного редактора могут сделать его сложным для освоения новичком? Основную трудность для новичков при освоении данного редактора могут составлять большое количество команд, комбинаций клавиш, которые не получится запомнить с

первого раза и поэтому придется часто обращаться к справочным материалам.

3. Своими словами опишите, что такое буфер и окно в терминологии emacs'a. 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 и 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. Можно ли открыть больше 10 буферов в одном окне? Да, можно.
5. Какие буферы создаются по умолчанию при запуске emacs? 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. Какие клавиши вы нажмёте, чтобы ввести следующую комбинацию C-c | и C-c C-|? C-c | сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши, нажимаю «|» C-c C-| сначала, удерживая «ctrl», нажимаю «с», после – отпускаю обе клавиши и, удерживая «ctrl», нажимаю «|».
7. Как поделить текущее окно на две части? Чтобы поделить окно на две части необходимо воспользоваться комбинацией «Ctrl-x 3» (по вертикали) или «Ctrl-x 2» (по горизонтали).
8. В каком файле хранятся настройки редактора emacs? Настройки Emacs хранятся в файле .emacs.
9. Какую функцию выполняет клавиша и можно ли её переназначить? По умолчанию клавиша «-» удаляет символ перед курсором, но в редакторе её можно переназначить. Для этого необходимо изменить конфигурацию файла .emacs.
10. Какой редактор вам показался удобнее в работе vi или emacs? Поясните почему. Более удобным я считаю редактор emacs, потому что Vim - это всего лишь редактор. Emacs - не только редактор, ко всему прочему это IDE, файловый менеджер, эмулятор терминала, веб-браузер, почтовый клиент, клиент новостей и много чего еще.