

# Отчёт по лабораторной работе №11

Операционные системы

---

Румянцев А. О.

20.04.2024

Российский университет дружбы народов, Москва, Россия

## Цель работы

---

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

## Задание

---

1. Ознакомиться с теоретическим материалом
2. Ознакомиться с редактором emacs.
3. Выполнить упражнения.
4. Ответить на контрольные вопросы.

## Теоретическое введение

---

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;

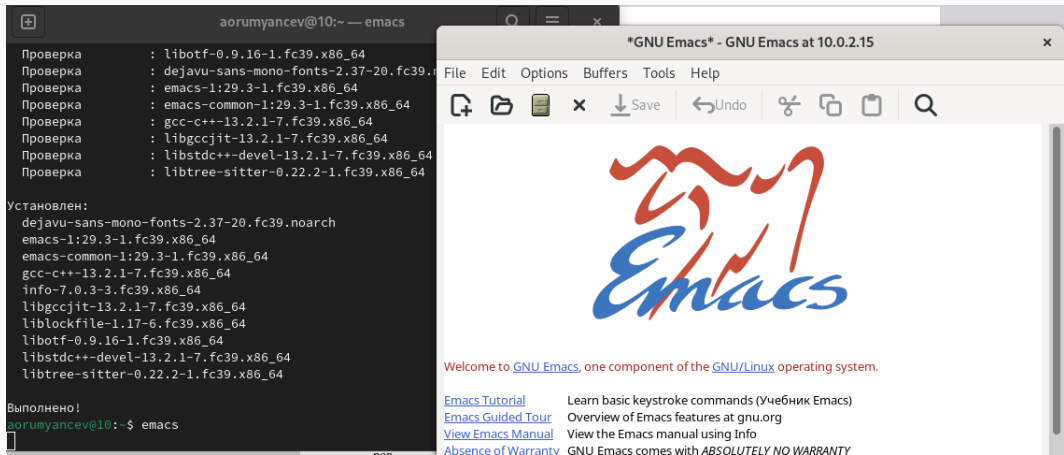
Всё это разнообразие достигается благодаря архитектуре Emacs, которая позволяет расширять возможности редактора при помощи языка Emacs Lisp. На языке C написаны лишь самые базовые и низкоуровневые части Emacs, включая полнофункциональный интерпретатор языка Lisp. Таким образом, Emacs имеет встроенный язык программирования, который может использоваться для настройки, расширения и изменения поведения редактора. В действительности, большая часть того редактора, с которым пользователи

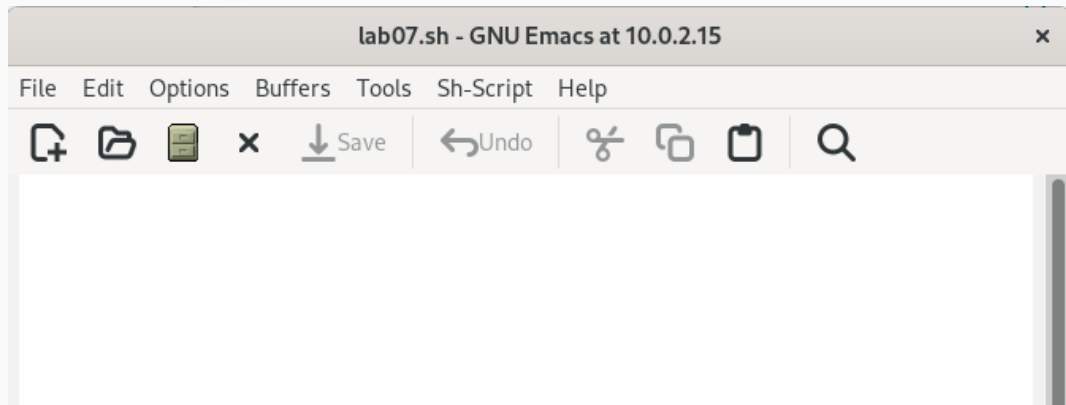
## Выполнение лабораторной работы

---

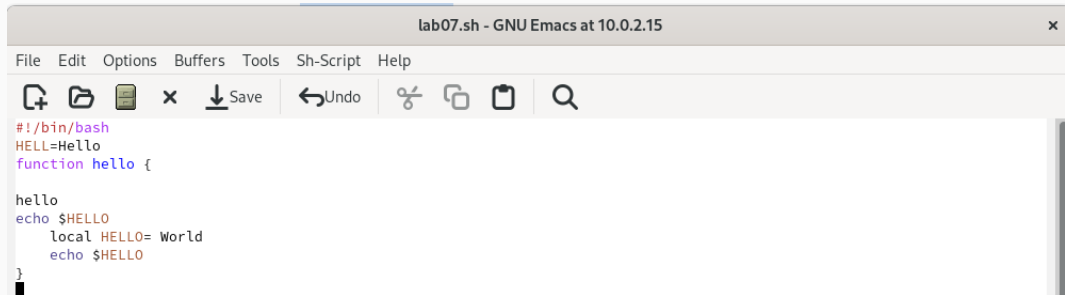


## Открываю Emacs через терминал.





Прописываю в файле текст программы.



The screenshot shows the GNU Emacs editor interface. The title bar at the top reads "lab07.sh - GNU Emacs at 10.0.2.15". Below the title bar is a menu bar with the following items: File, Edit, Options, Buffers, Tools, Sh-Script, and Help. Below the menu bar is a toolbar with icons for opening a file, saving a file, closing a file, saving the current file, undo, redo, copy, paste, and search. The main editing area contains the following text:

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

Сохраняю файл с помощью комбинации C-x C-s.

```
U:--- lab07.sh All L8 (Shell-script[sh])  
Wrote /home/aorumyancev/work/study/2023-2024/OS/os-intro/labs/lab11/report/lab07  
$ .sh
```

## Вырезаю одной командой целую строку (C-к)

```
    echo $HELLO  
}  
  
hello
```

```
U:**- lab07.sh      All  L7      (Shell-script[sh])
```

Вставляю эту строку в конец файла (С-у).

```
    echo $HELLO  
}  
  
hello  
echo $HELLO
```

```
U:**- lab07.sh      All  L9      (Shell-script[sh])
```

```
Mark set
```

Выделяю область текста(C-space), копирую область в буфер обмена (M-w), вырезаю эту область с помощью C-w.

Вставляю область в конец файла (C-y).

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

hello
echo $HELLO
    LOCAL HELLO=World
    echo $HELLO
}
```

U:\*\*- lab07.sh All L10 (Shell-script[sh])

Mark set



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

hello
echo $HELLO
█
```

U:\*\*\* lab07.sh All L7 (Shell-script[sh])

Undo

## Перевожу курсор в начало строки C-a

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

hello
echo $HELLO
    LOCAL HELLO=World
    echo $HELLO
}
```

U:~ - lab07.sh All L7 (Shell-script[sh])

## Перемещаю курсор в конец строки C-e

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

hello
echo $HELLO
    LOCAL HELLO=World
    echo $HELLO
}
```

U:~\*- lab07.sh All L7 (Shell-script[sh])

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

hello
echo $HELLO
    LOCAL HELLO=World
    echo $HELLO
}
```

U:~\*- lab07.sh All L1 (Shell-script[sh])

Перемещаю курсор в конец файла M->

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

hello
echo $HELLO
    LOCAL HELLO=World
    echo $HELLO
}
```

U:~ - lab07.sh All L9 (Shell-script[sh])

## Открываю список активных буферов в другой окне C-x C-b

U:***- lab07.sh Bot L9 (Shell-script[sh])				
CRM	Buffer	Size	Mode	File
[.]	* lab07.sh	101	Shell-script[sh]	~/work/study/2023-2024/OS/os-in
%	*GNU Emacs*	734	Fundamental	
	*scratch*	145	Lisp Interaction	
%%	*Messages*	1176	Messages	
%%	*Async-native-compile...	165	Fundamental	
U:%%- *Buffer List* All L1 (Buffer Menu)				
C-x C-b				

```
U:**- lab07.sh      All   L13   (Shell-script[sh])
```

```
#!/bin/bash
```

```
HELL=Hello
```

```
function hello {
```

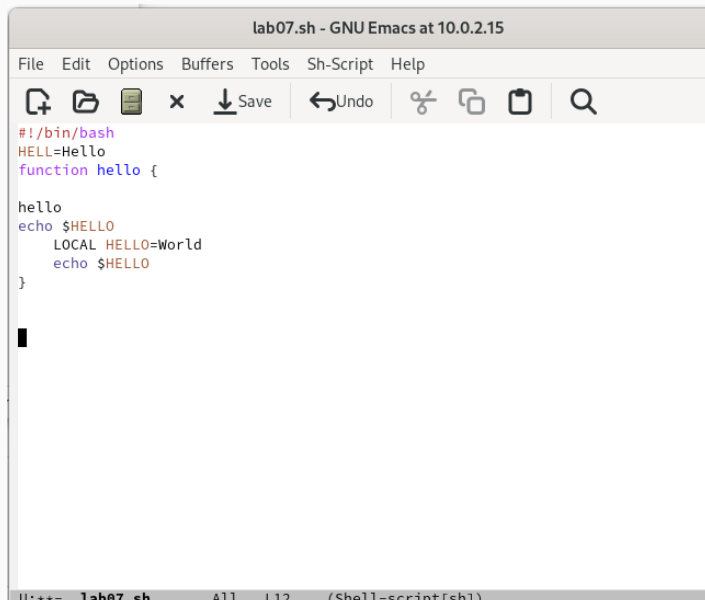
```
hello
```

```
echo $HELLO
```

```
    LOCAL HELLO=World
```

```
U:**- lab07.sh      Top    L7    (Shell-script[sh])
```

## Закрываю окно другого буфера C-x 0



```
lab07.sh - GNU Emacs at 10.0.2.15
File Edit Options Buffers Tools Sh-Script Help
[Icons: Open, Save, Undo, Redo, Search]

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


```

lab07.sh All 112 (Shell-script[sh])

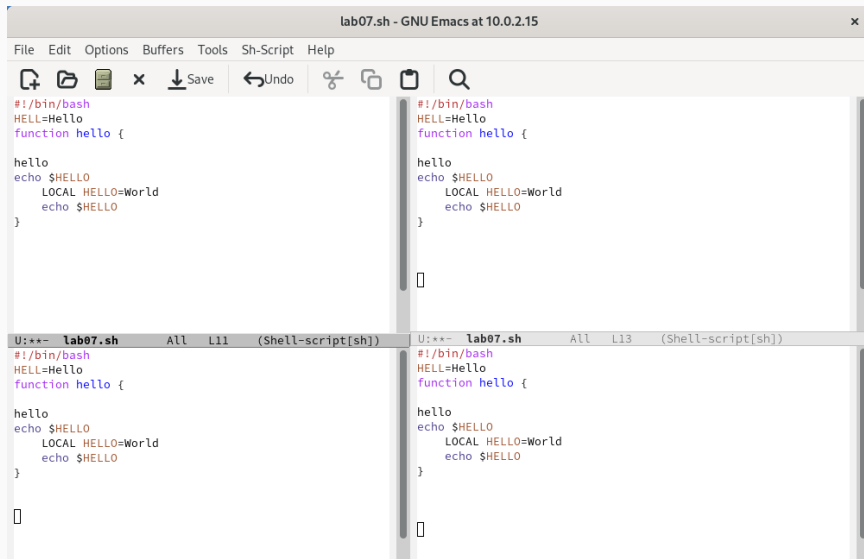


Открываю другой буфер без вывода их списка на экран с помощью C-x b

```
U:**- lab07.sh      All  L15  (Shell-script[sh])
CRM Buffer          Size Mode      File
. * lab07.sh        103 Shell-script[sh] ~/work/study/2023-2024/OS/os-i
%* *Messages*       1605 Messages
%  *GNU Emacs*      734 Fundamental
   *scratch*        145 Lisp Interaction
%* *Async-native-compile-... 165 Fundamental

U:%%- *Buffer List* All  L1  (Buffer Menu)
Switch to buffer (default *Messages*): Messages
```

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



```
lab07.sh - GNU Emacs at 10.0.2.15
File Edit Options Buffers Tools Sh-Script Help
[Icons] Save Undo [Icons] Search

#!/bin/bash
HELL=Hello
function hello {

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

[ ]

U:*** lab07.sh All L11 (Shell-script[sh])
#!/bin/bash
HELL=Hello
function hello {

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

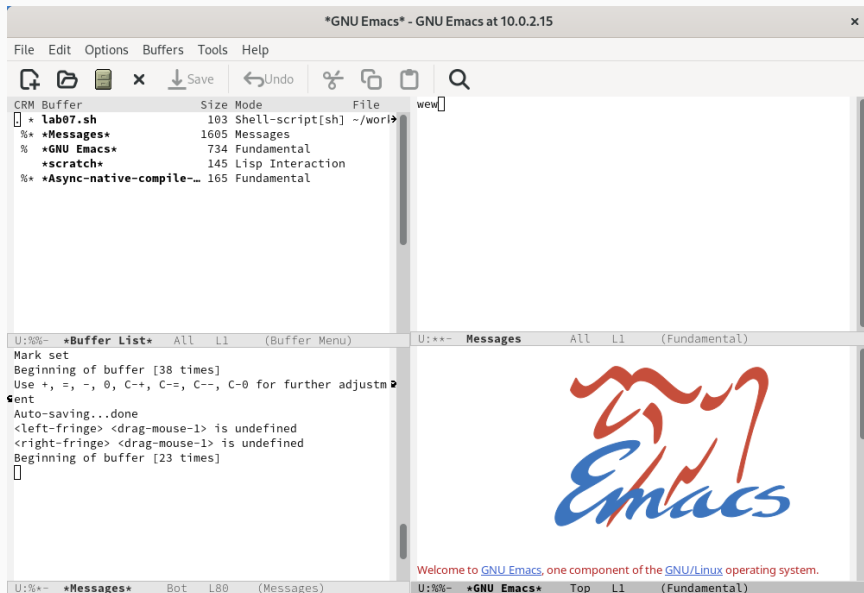
[ ]

U:*** lab07.sh All L13 (Shell-script[sh])
#!/bin/bash
HELL=Hello
function hello {

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

[ ]
```

В каждой из четырех созданных окон с помощью C-x b открыл разные буферы.



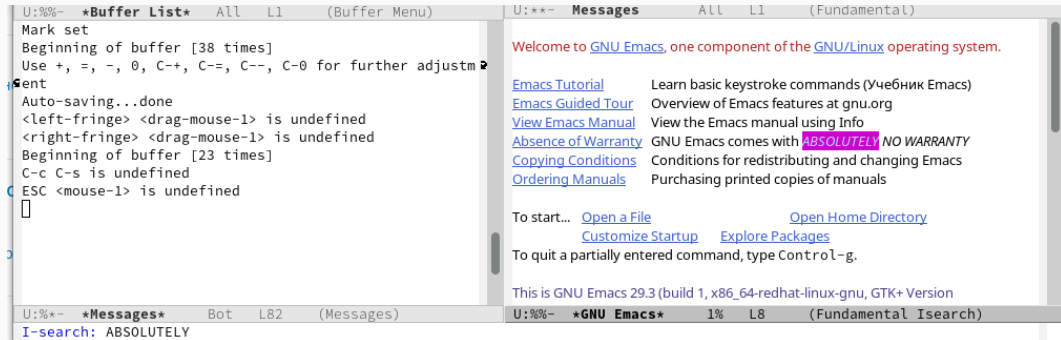
Перехожу в режим поиска с помощью C-s, ищу слова в тексте, они подсвечиваются.

```
U:%%- *Buffer List* All L1 (Buffer Menu)
Mark set
Beginning of buffer [38 times]
Use +, =, -, 0, C-+, C-=, C--, C-0 for further adjustm
Sent
Auto-saving...done
<left-fringe> <drag-mouse-1> is undefined
<right-fringe> <drag-mouse-1> is undefined
Beginning of buffer [23 times]

C
D

U:%%*- *Messages* Bot L80 (Messages Isearch)
Failing I-search: left
```

С помощью той же комбинации C-s я могу перемещаться по результатам поиска.



The screenshot shows the Emacs interface with two buffers visible. The left buffer, titled `*Buffer List*`, contains a list of buffers and their contents. The right buffer, titled `Messages`, displays the Emacs welcome message and a list of links for learning more about Emacs. The status bar at the bottom shows the current buffer is `*Messages*` and the search term is `ABSOLUTELY`.

```
U:%%- *Buffer List* All L1 (Buffer Menu)
Mark set
Beginning of buffer [38 times]
Use +, =, -, 0, C-+, C-=, C--, C-0 for further adjustm
Sent
Auto-saving...done
<left-fringe> <drag-mouse-1> is undefined
<right-fringe> <drag-mouse-1> is undefined
Beginning of buffer [23 times]
C-c C-s is undefined
ESC <mouse-1> is undefined
[]

U:%%- *Messages* All L1 (Fundamental)
Welcome to GNU Emacs, one component of the GNU/Linux operating system.

Emacs Tutorial      Learn basic keystroke commands (Учебник Emacs)
Emacs Guided Tour   Overview of Emacs features at gnu.org
View Emacs Manual   View the Emacs manual using Info
Absence of Warranty GNU Emacs comes with ABSOLUTELY NO WARRANTY
Copying Conditions  Conditions for redistributing and changing Emacs
Ordering Manuals    Purchasing printed copies of manuals

To start... Open a File Open Home Directory
Customize Startup Explore Packages

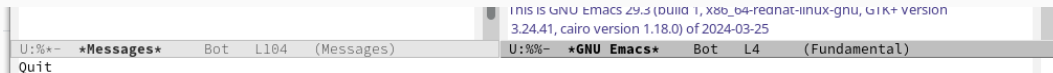
To quit a partially entered command, type Control-g.

This is GNU Emacs 29.3 (build 1, x86_64-redhat-linux-gnu, GTK+ Version

U:%%- *Messages* Bot L82 (Messages)
I-search: ABSOLUTELY

U:%%- *GNU Emacs* 1% L8 (Fundamental Isearch)
```

С помощью C-g выхожу из режима поиска, снимается выделение.



The screenshot shows two Emacs windows. The left window, titled '\*Messages\*', displays search results for 'Bot' on line 104 of 'Messages'. The right window, titled '\*GNU Emacs\*', shows the Emacs version information: 'This is GNU Emacs 29.3 (build 1, x86\_64-redhat-linux-gnu, GTK+ version 3.24.41, cairo version 1.18.0) of 2024-03-25'. Both windows have a status bar at the bottom indicating the current buffer and line.

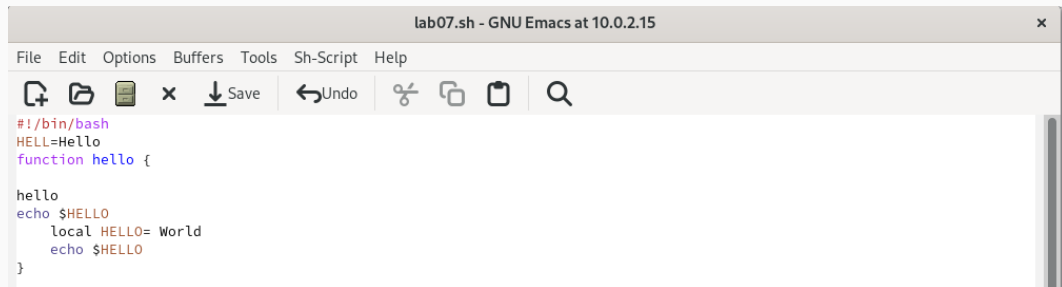
```
U:%*- *Messages* Bot L104 (Messages) U:%%- *GNU Emacs* Bot L4 (Fundamental)
```

Quit

Перехожу в режим поиска и замены с помощью M-%, ввожу какое слово хочу заменить, затем ввожу на какое хочу заменить

```
U:**~ lab07.sh All L14 (Shell-script[sh])  
Query replace LOCAL with: local
```

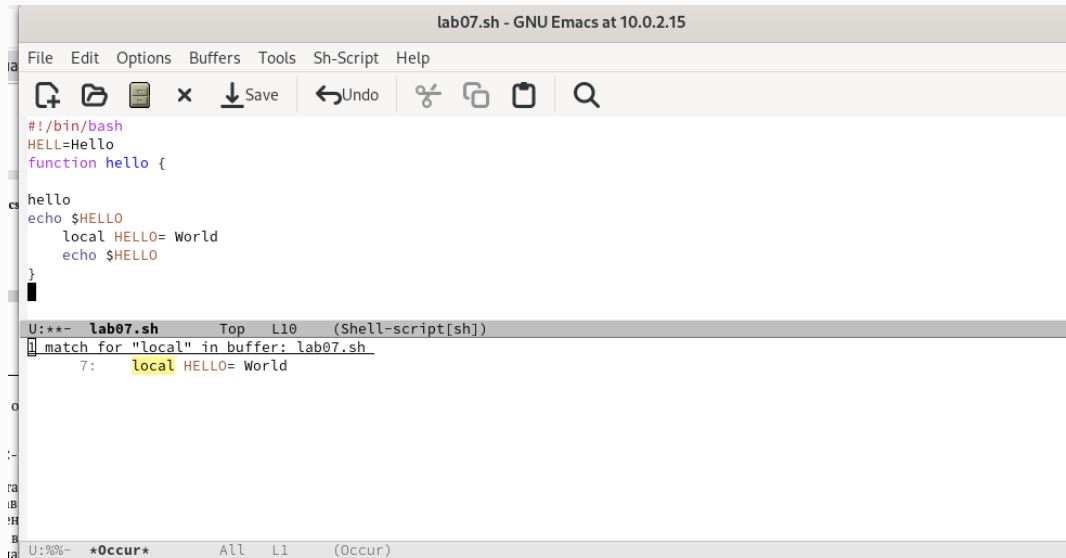
Видим, что слова были заменены успешно

A screenshot of a GNU Emacs editor window titled "lab07.sh - GNU Emacs at 10.0.2.15". The window has a menu bar with "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". Below the menu bar is a toolbar with icons for opening, saving, undo, redo, copy, paste, and search. The main text area contains a shell script with the following content:

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



С помощью M-s о перехожу в другой режим поиска. Он отличается от предыдущего тем, что выводит результат в отдельном окне от окна буфера.



The screenshot shows the GNU Emacs editor interface. The title bar reads "lab07.sh - GNU Emacs at 10.0.2.15". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", and "Help". The toolbar contains icons for file operations (new, open, save, close), editing (undo, redo, cut, copy, paste), and search (find). The main buffer contains a shell script:

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

```

Below the script, a status line shows "U:\*\*\* lab07.sh Top L10 (Shell-script[sh])". A search results window is open, displaying the command "match for 'local' in buffer: lab07.sh" and the result "7: local HELLO= World". The bottom status bar shows "U:%%- \*Occur\* All L1 (Occur)".

## Выводы

---

В ходе данной лабораторной работы я познакомился с ос Linux, получил практические навыки работы с редактором Emacs.

## Ответы на контрольные вопросы

---

## 1. Кратко охарактеризуйте редактор emacs.

Emacs — один из наиболее мощных и широко распространённых редакторов, используемых в мире UNIX. Написан на языке высокого уровня Lisp.

## 2. Какие особенности данного редактора могут сделать его сложным для освоения новичком?

Большое разнообразие сложных комбинаций клавиш, которые необходимы для редактирования файла и в принципе для работа с Emacs.

### 3. Своими словами опишите, что такое буфер и окно в терминологии emacs'а.

Буфер - это объект в виде текста. Окно - это прямоугольная область, в которой отображен буфер.

#### 4. Можно ли открыть больше 10 буферов в одном окне?

Да, можно.



## 5. Какие буферы создаются по умолчанию при запуске emacs?

Emacs использует буферы с именами, начинающимися с пробела, для внутренних целей. Отчасти он обращается с буферами с такими именами особым образом — например, по умолчанию в них не записывается информация для отмены изменений.

6. Какие клавиши вы нажмёте, чтобы ввести следующую комбинацию C-с | и C-с C-|?

Ctrl + c, а потом | и Ctrl + c Ctrl + |

## 7. Как разделить текущее окно на две части?

С помощью команды `Ctrl + x 3` (по вертикали) и `Ctrl + x 2` (по горизонтали).

## 8. В каком файле хранятся настройки редактора emacs?

Настройки emacs хранятся в файле `.emacs`, который хранится в домашней директории пользователя. Кроме этого файла есть ещё папка `.emacs`.

## 9. Какую функцию выполняет клавиша и можно ли её переназначить?

Выполняет функцию стереть, думаю можно переназначить.

10. Какой редактор вам показался удобнее в работе vi или emacs? Поясните почему.

Для меня удобнее был редактор Emacs, так как у него есть командная оболочка. А vi открывается в терминале, и выглядит своеобразно.