

# Отчёт по лабораторной работе №11

## Текстовый редактор emacs

Мутале Чали

### Содержание

Цель работы .....	1
Задание .....	1
Выполнение лабораторной работы .....	1
Выводы .....	9
Ответы на контрольные вопросы .....	9

### Цель работы

Получить практические навыки работы с редактором Emacs.

### Задание

1. Выполнить основные команды emacs

### Выполнение лабораторной работы

Для данной работы, мне надо был установить Emacs:

```
cmutale@cmutale:~$ sudo dnf install emacs
Updating and loading repositories:
Copr repo for gitflow owned by elegos
```

*Рис 1: Установка Emacs*

Выполнив Emacs в командной строке, я открыл текстовый редактор:

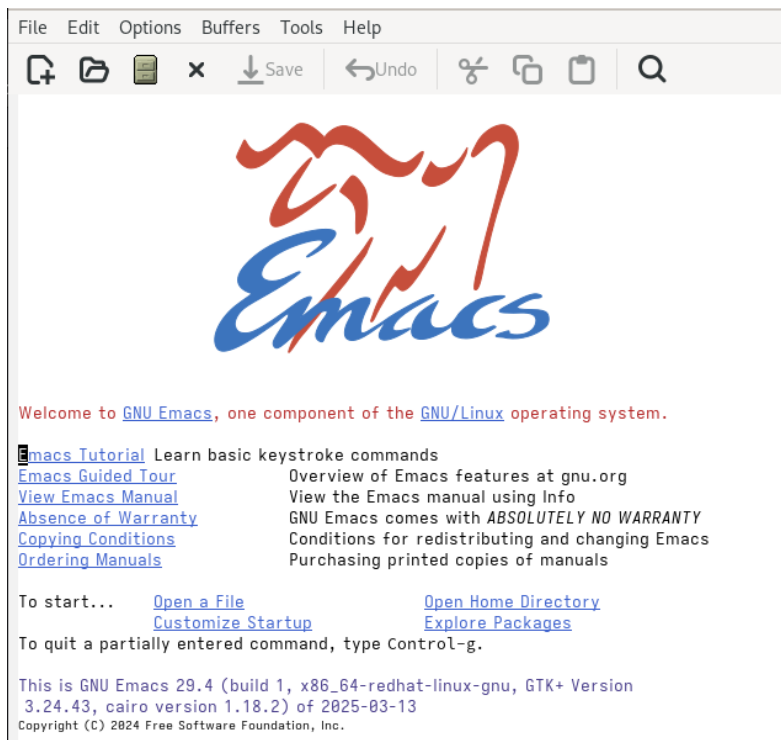


Рис 2: Emacs

С помощью комбинации Ctrl-x Ctrl-f, создал файл lab07.sh:

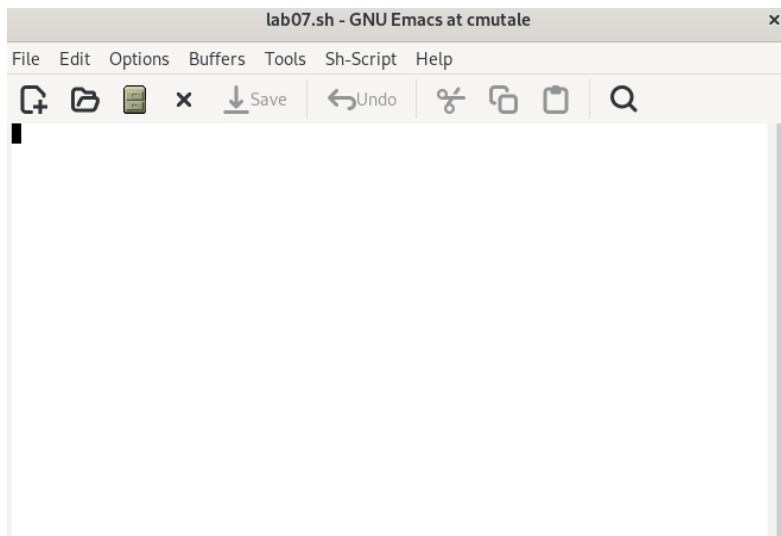
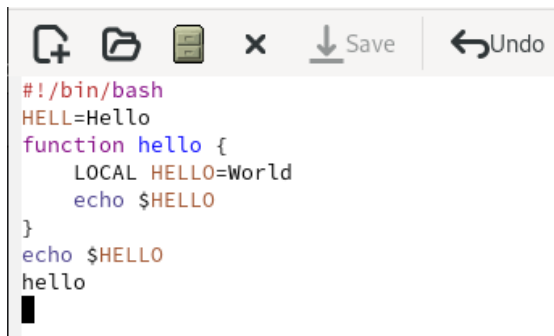


Рис 3: Созданный файл

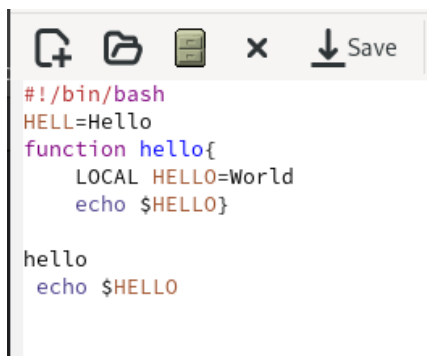
Я написал некоторый текст в этом же файле (lab07.sh). После этого сохранил файл с помощью комбинации Ctrl-x Ctrl-s:



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

Рис 4: текст в lab07.sh

Одной командой вырезал целую строку (C-k):

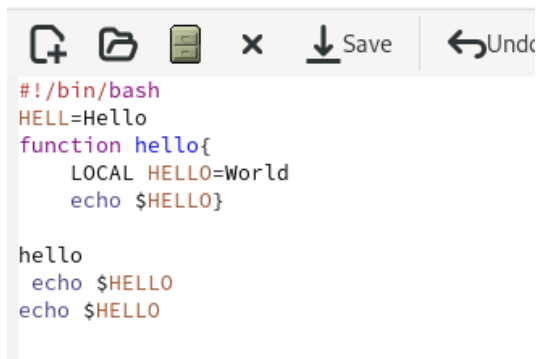


```
#!/bin/bash
HELL=Hello
function hello{
    LOCAL HELLO=World
    echo $HELLO}

hello
echo $HELLO
```

Рис 5: Вырезание строки

С помощью C-u вставил эту строку в конец файла:

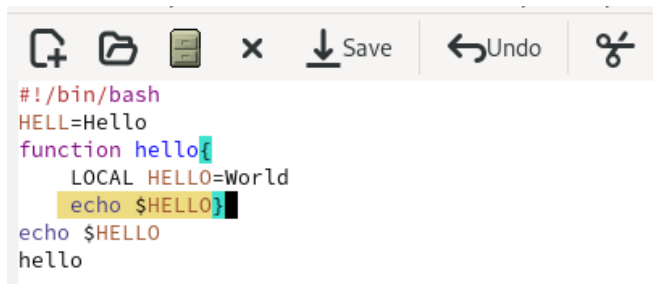


```
#!/bin/bash
HELL=Hello
function hello{
    LOCAL HELLO=World
    echo $HELLO}

hello
echo $HELLO
echo $HELLO
```

Рис 6: Перемещение строку в конец файла

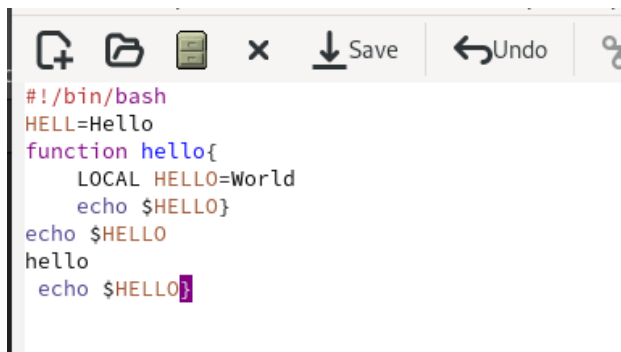
Выделил область текста (C-space):



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

Рис 7: Выделенный текст

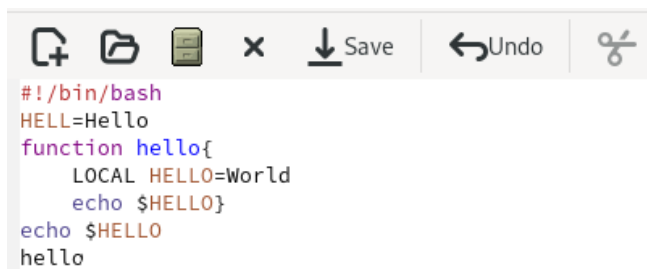
Скопировал область в буфер обмена (M-w) и вставил ее в конец файла:



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

Рис 8: копирование и вставка

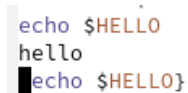
Выделил эту же область и на этот раз вырезал её (C-w):



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

Рис 9: Вырезанная область

С помощью C-/ отменил последнее действие:



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


Рис 10: отмена действие

С помощью C-a переместил курсор в начало строки:

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

Рис 11: Перемещение курсор в начало строки

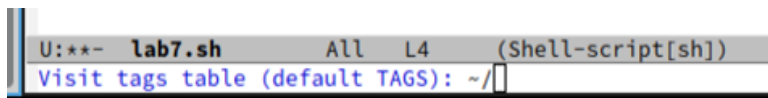
С помощью C-е переместил курсор в конец строки:



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

Рис 12: Перемещение курсор в конец строки

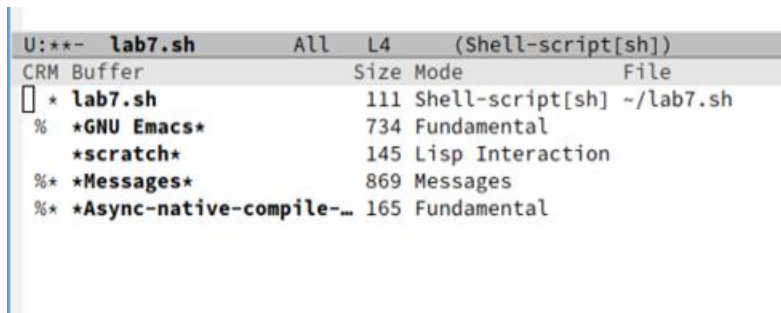
Переместил курсор в начало и конец буфера с помощью M-< и M-> соответственно:



```
U:*** lab7.sh All L4 (Shell-script[sh])
Visit tags table (default TAGS): ~/
```

Рис 13: Перемещение курсор в буфере

Выводил список активных буферов на экран с помощью C-x C-b:



```
U:*** lab7.sh All L4 (Shell-script[sh])
CRM Buffer Size Mode File
* lab7.sh 111 Shell-script[sh] ~/lab7.sh
% *GNU Emacs* 734 Fundamental
*scratch* 145 Lisp Interaction
%* *Messages* 869 Messages
%* *Async-native-compile-... 165 Fundamental
```

Рис 14: Активные буферы

С помощью C-x o переместился во вновь открытое окно со списком открытых буферов и переключился на другой буфер:

```
--:--- lab7.sh All L9 (Shell-script[bash])
CRM Buffer Size Mode File
| lab7.sh 111 Shell-script[ba... ~/lab7.sh
% *GNU Emacs* 734 Fundamental
% *scratch* 145 Lisp Interaction
%* *Messages* 552 Messages
%* *Async-native-compile-... 165 Fundamental

U:%%- *Buffer List* All L1 (Buffer Menu)
```

Рис 15: список открытых буферов

С помощью C-x 0 закрыл окно со списком открытых буферов:

```
--:--- lab7.sh All L9 (Shell-script[bash])
C-x 0
```

Рис 16: Закрытие окно

Без вывода списка буферов, я переключился между буферами:

```
--:--- lab7.sh All L9 (Shell-script[bash])
Switch to buffer (default b): |
```

Рис 17: Переключение между буферами

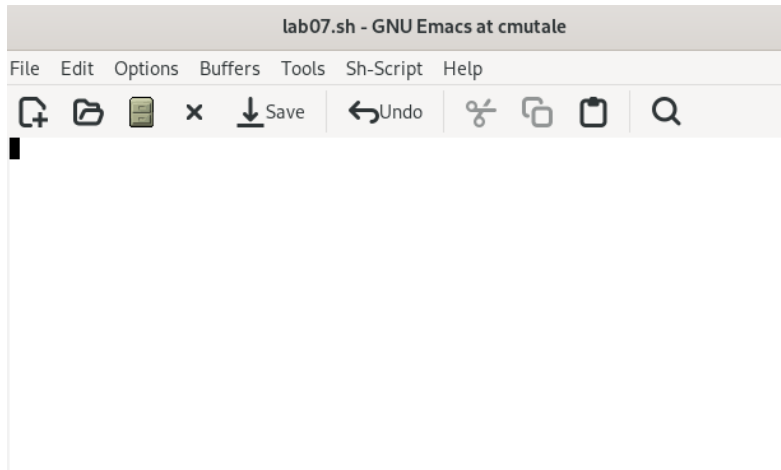


Рис 18: Новый буфер

Поделил фрейм на 4 части. Сначала я разделил фрейм на два окна по вертикали (С-х 3), а затем каждое из этих окон на две части по горизонтали (С-х 2):

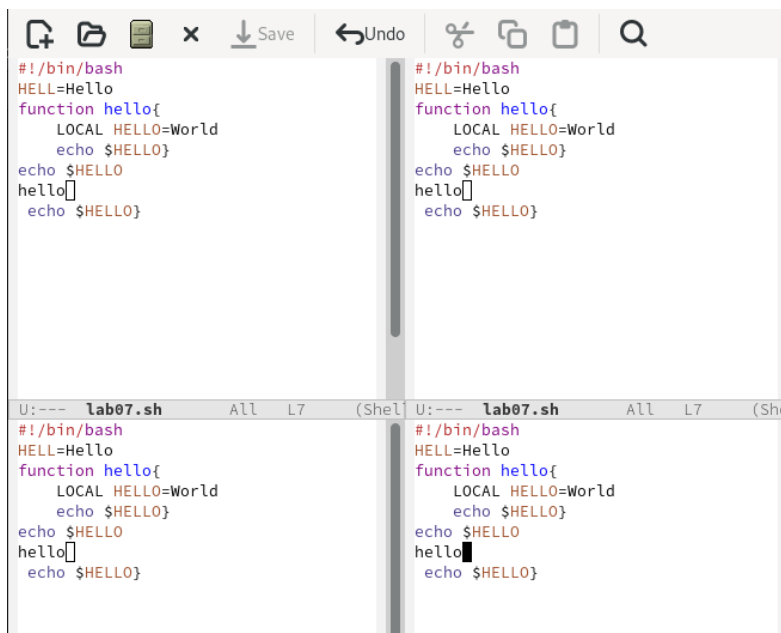


Рис 19: Фрейм разделённый на 4 окна

В каждом из четырёх созданных окон открыл новый буфер:

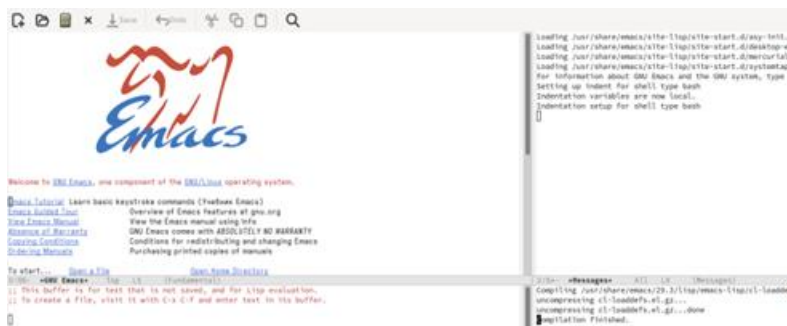


Рис 20: Новые буферы

Переключился в режим поиска (C-s) и искала Indent:

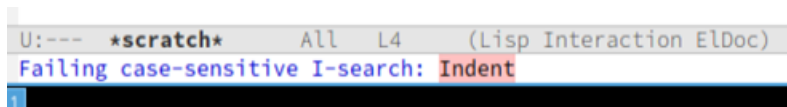


Рис 21: Режим поиска

Переключался между результатами поиска, нажимая C-s и вышла из режима поиска, нажав C-g:

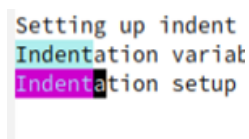


Рис 22: Переключение между результатами

Перешёл в режим поиска и замены (M-%), искал слово World, нажал Enter, и заменил на Planet:

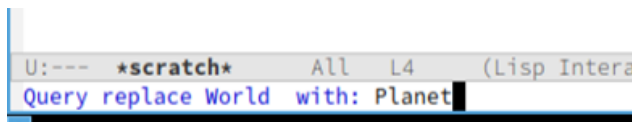


Рис 23: Режим поиска

Нажав M-s o, я использовал другой режим поиска. Он отличается от предыдущего тем, что выводит результаты поиска в новом окне:

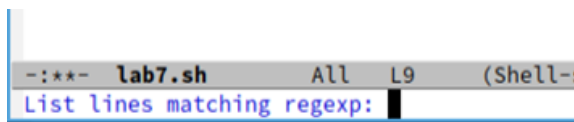


Рис 24: другой режим поиска



A screenshot of the Emacs editor showing search results for the string "hello" in the buffer "lab7.sh". The search results are displayed in a list format, with line numbers and the corresponding code lines. The lines are: 2:HELL=Hello, 3:function hello{, 4: LOCAL HELLO=Planet, 5: echo \$HELLO}, 7:hello, and 8: echo \$HELLO. The search results are highlighted in yellow.

Рис 25: Результаты поиска

## Выводы

При выполнении данной работы я получил практические навыки работы с Emacs.

## Ответы на контрольные вопросы

1. Emacs — один из наиболее мощных и широко распространённых редакторов, используемых в мире UNIX. Написан на языке высокого уровня Lisp.
2. Большое разнообразие сложных комбинаций клавиш, которые необходимы для редактирования файла и в принципе для работы с Emacs.
3. Буфер - это объект в виде текста. Окно - это область, в которой отображен буфер.
4. Да, можно.
5. 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. Ctrl + c, а потом | и Ctrl + c Ctrl + |
7. С помощью команды Ctrl + x 3 (по вертикали) и Ctrl + x 2 (по горизонтали).
8. Настройки emacs хранятся в файле .emacs, который хранится в домашней директории пользователя. Кроме этого файла есть ещё папка .emacs.
9. Выполняет функцию стереть, думаю можно переназначить.
10. Для меня удобнее был редактор Emacs, так как у него есть командная оболочка. А vi открывается в терминале, и выглядит своеобразно.