

# **Отчёт по лабораторной работе №9**

**Текстовый редактор etacs**

Грузинова Елизавета Константиновна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
<b>5</b>	<b>Контрольные вопросы</b>	<b>22</b>
<b>6</b>	<b>Выводы</b>	<b>24</b>
	<b>Список литературы</b>	<b>25</b>

## Список иллюстраций

4.1	Команда для вызова Emacs . . . . .	8
4.2	Создание файла в Emacs . . . . .	9
4.3	Файл lab07.sh . . . . .	9
4.4	Вырезка строки echo \$HELLO . . . . .	10
4.5	Добавление echo \$HELLO в конец файла . . . . .	11
4.6	Mark set . . . . .	12
4.7	Вставка из буфера обмена . . . . .	13
4.8	Удаление области . . . . .	14
4.9	Отмена предыдущего шага . . . . .	15
4.10	Курсор в начале строки . . . . .	15
4.11	Курсор в конце строки . . . . .	16
4.12	Курсор в начале буфера . . . . .	16
4.13	Курсор в конце буфера . . . . .	16
4.14	Вывод буферов на экран . . . . .	16
4.15	Переключение между буферами . . . . .	17
4.16	Переключение между буферами без их вывода на экран . . . . .	17
4.17	Разделение emacs на 4 активных окна . . . . .	18
4.18	Создание новых файлов в каждом соответствующем окне . . . . .	18
4.19	Поиск e в строке . . . . .	19
4.20	Переключение между результатами поиска . . . . .	19
4.21	Режим поиска и замены через панель edit . . . . .	20
4.22	Изменяемое слова . . . . .	20
4.23	Замена слова World на Mir . . . . .	20
4.24	Режим отличается тем, что создает дополнительную вкладку с текстом, в котором находится предмет поиска . . . . .	21

# 1 Цель работы

Познакомиться с операционной системой Linux и получить практические навыки работы с редактором Emacs.

## 2 Задание

1. Ознакомиться с теоретическим материалом.
2. Ознакомиться с редактором emacs.
3. Выполнить упражнения.
4. Открыть emacs.
5. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).
6. Наберите заданный в лабораторной работе текст.
7. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).
8. Прodelать с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш.
  - 5.1. Вырезать одной командой целую строку (C-k).
  - 5.2. Вставить эту строку в конец файла (C-y).
  - 5.3. Выделить область текста (C-space).
  - 5.4. Скопировать область в буфер обмена (M-w).
  - 5.5. Вставить область в конец файла.
  - 5.6. Вновь выделить эту область и на этот раз вырезать её (C-w).
  - 5.7. Отмените последнее действие (C-/).
6. Научитесь использовать команды по перемещению курсора.

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

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

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

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

7. Управление буферами.

7.1. Вывести список активных буферов на экран (C-x C-b).

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

7.3. Закройте это окно (C-x 0).

7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b).

8. Управление окнами.

8.1. Поделите 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).

8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста.

9. Режим поиска.

9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте.

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

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

9.4. Перейдите в режим поиска и замены (M-%), введите текст, который следует найти и заменить, нажмите Enter, затем введите текст для замены. После того как будут подсвечены результаты поиска, нажмите ! для подтверждения замены.

9.5. Испробуйте другой режим поиска, нажав M-s o. Объясните, чем он отличается от обычного режима?

4. Ответить на контрольные вопросы.

## 3 Теоретическое введение

Emacs - один из двух наиболее широко используемых текстовых редакторов на платформах Unix и Linux.

Emacs - это больше, чем просто текстовый редактор; он также может выдавать команды оболочки, выходить в Интернет, писать и тестировать программы, а также читать и отправлять электронные письма в среде Emacs.

Еще одна полезная функция, которая делает текст и код более читабельным, - это автоматическое отступление. Emacs идентифицирует блоки кода и группирует их соответственно. Он также обеспечивает поддержку символов Unicode практически для всех систем письма и языков, а также обеспечивает самодокументирование, которое автоматически генерирует и отображает документацию для каждой команды, переменной и внутренней функции, найденной в исходном коде программы. [1]

## 4 Выполнение лабораторной работы

Выполнить упражнения.

1. Открыть emacs.(рис. 4.1)



```
[ekgruzinova@fedora ~]$ emacs
```

Рис. 4.1: Команда для вызова Emacs

2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).(рис. 4.2)



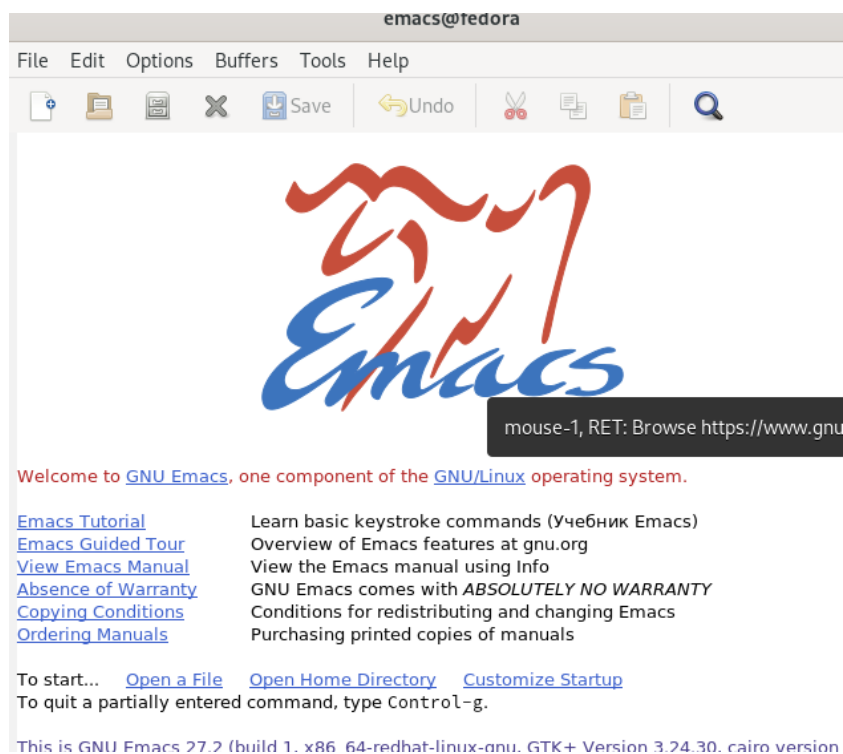


Рис. 4.2: Создание файла в Emacs

3. Наберите заданный в лабораторной работе текст. 4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).(рис. 4.3)

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

Рис. 4.3: Файл lab07.sh

5. Прodelать с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш.

5.1. Вырезать одной командой целую строку (C-k).(рис. 4.4)



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

Рис. 4.4: Вырезка строки echo \$HELLO

5.2. Вставить эту строку в конец файла (C-y).(рис. 4.5)

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

Рис. 4.5: Добавление echo \$HELLO в конец файла

5.3. Выделить область текста (C-space).(рис. 4.6)

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

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

Рис. 4.6: Mark set

5.4. Скопировать область в буфер обмена (M-w). 5.5. Вставить область в конец файла.(рис. 4.7)

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

Рис. 4.7: Вставка из буфера обмена

5.6. Вновь выделить эту область и на этот раз вырезать её (C-w).(рис. 4.8)

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

Рис. 4.8: Удаление области

5.7. Отмените последнее действие (C-/).(рис. 4.9)

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

Рис. 4.9: Отмена предыдущего шага

6. Научитесь использовать команды по перемещению курсора.

6.1. Переместите курсор в начало строки (C-a).(рис. 4.10)

```
echo $HELLO  
hello
```

Рис. 4.10: Курсор в начале строки

6.2. Переместите курсор в конец строки (C-e).(рис. 4.11)



Рис. 4.11: Курсор в конце строки

6.3. Переместите курсор в начало буфера (M-<).(рис. 4.12)

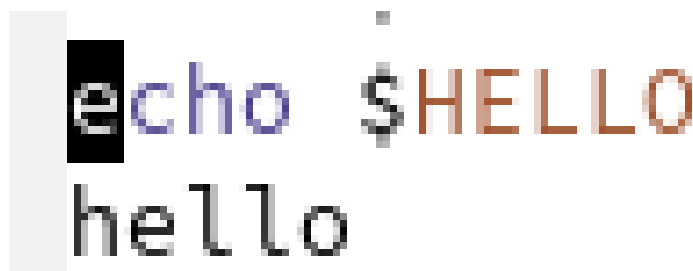


Рис. 4.12: Курсор в начале буфера

6.4. Переместите курсор в конец буфера (M->).(рис. 4.13)



Рис. 4.13: Курсор в конце буфера

## 7. Управление буферами.

7.1. Вывести список активных буферов на экран (C-x C-b). (рис. 4.14)

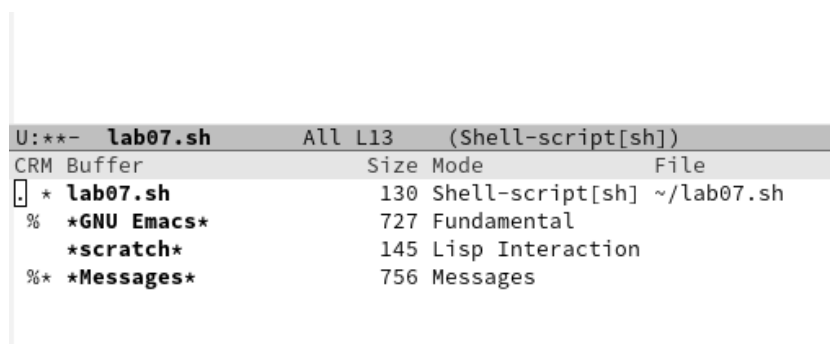


Рис. 4.14: Вывод буферов на экран



7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых буферов и переключитесь на другой буфер. (рис. 4.15)

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

CRM	Buffer	Size	Mode	File
	lab07	0	Fundamental	
	*scratch*	145	Lisp Interaction	
%	*Help*	215	Help	
%	*GNU Emacs*	727	Fundamental	
.	*lab07.sh	130	Shell-script[sh]	~/lab07.sh
%*	*Messages*	1149	Messages	

Рис. 4.15: Переключение между буферами

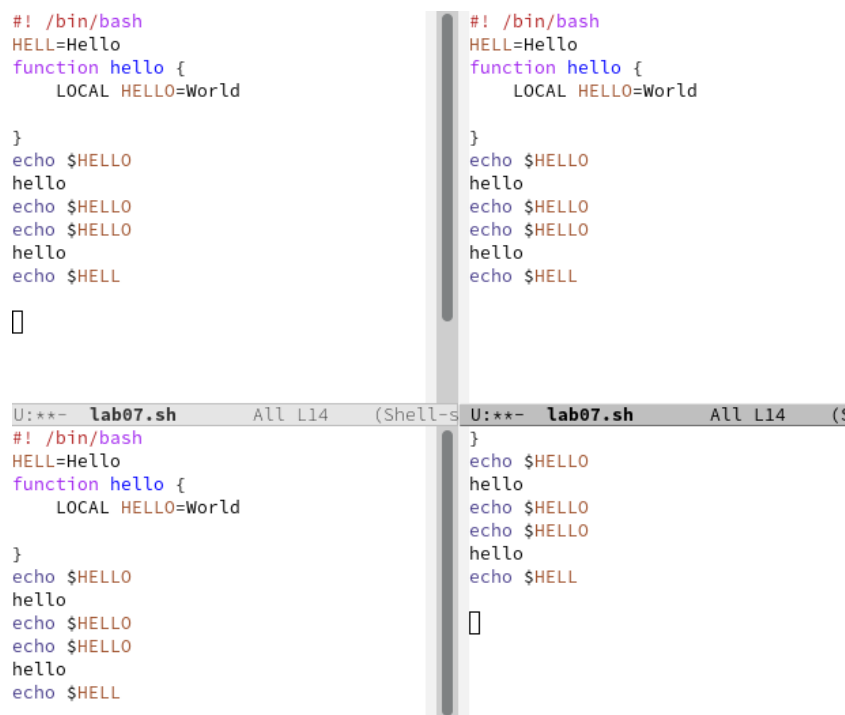
7.3. Закройте это окно (C-x 0). 7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b). (рис. 4.16)

```
U: **~ lab07.sh All L12 (Shell-s
Switch to buffer (default *scratch*):
```

Рис. 4.16: Переключение между буферами без их вывода на экран

## 8. Управление окнами.

8.1. Поделите фрейм на 4 части: разделите фрейм на два окна по вертикали (С-х 3), а затем каждое из этих окон на две части по горизонтали (С-х 2). (рис. 4.17)



```

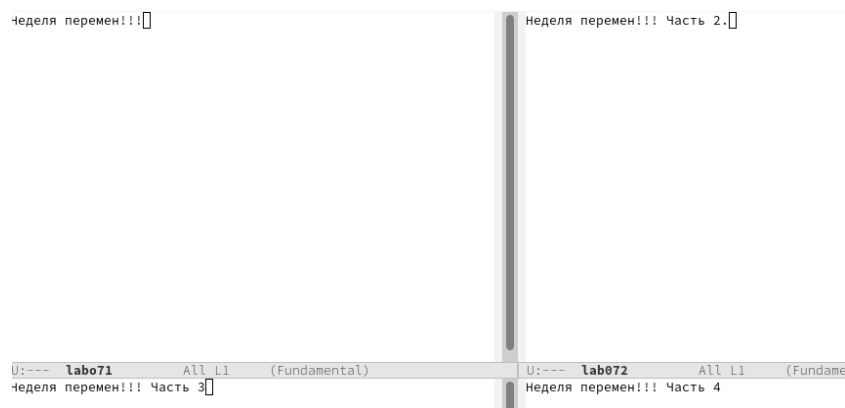
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
}
echo $HELLO
hello
echo $HELLO
echo $HELLO
hello
echo $HELL

U:*** lab07.sh All L14 (Shell-s
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO=World
}
echo $HELLO
hello
echo $HELLO
echo $HELLO
hello
echo $HELL

U:*** lab07.sh All L14 (
}
echo $HELLO
hello
echo $HELLO
echo $HELLO
hello
echo $HELL
  
```

Рис. 4.17: Разделение emacs на 4 активных окна

8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста. (рис. 4.18)



```

Неделя перемен!!!
Неделя перемен!!! Часть 2.
Неделя перемен!!! Часть 3.
Неделя перемен!!! Часть 4.
  
```

Рис. 4.18: Создание новых файлов в каждом соответствующем окне

## 9. Режим поиска.

9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте. (рис. 4.19)

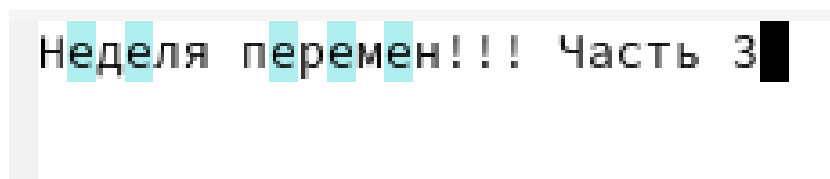


Рис. 4.19: Поиск е в строке

9.2. Переключайтесь между результатами поиска, нажимая C-s. 9.3. Выйдите из режима поиска, нажав C-g. (рис. 4.20)



Рис. 4.20: Переключение между результатами поиска

9.4. Перейдите в режим поиска и замены (M-%), введите текст, который следует найти и заменить, нажмите Enter, затем введите текст для замены. После того как будут подсвечены результаты поиска, нажмите ! для подтверждения замены. (рис. 4.21, 4.22, 4.23)

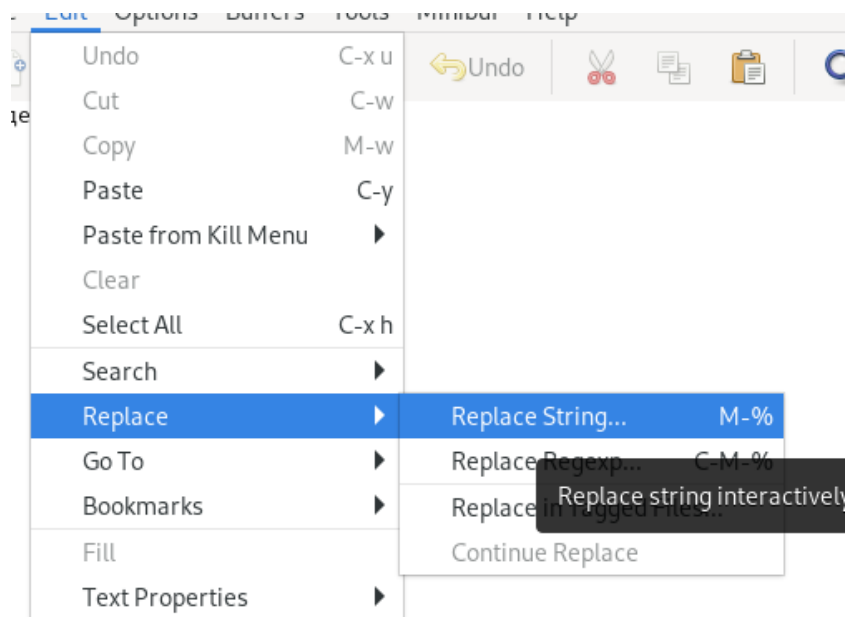


Рис. 4.21: Режим поиска и замены через панель edit

```
Неделя перемен!!! Часть 3
Hello,
World
```

Рис. 4.22: Изменяемое слова

```
Неделя перемен!!! Часть 3
Hello,
Mir
```

Рис. 4.23: Замена слова World на Mir

9.5. Испробуйте другой режим поиска, нажав M-s o. Объясните, чем он отличается от обычного режима? (рис. 4.24)

```
:**- lab073 All L3 (Fundamental)  
match for "3" in buffer: lab073  
1:Неделя перемен!!! Часть 3
```

Рис. 4.24: Режим отличается тем, что создает дополнительную вкладку с текстом, в котором находится предмет поиска

## 5 Контрольные вопросы

1. Кратко охарактеризуйте редактор emacs.

Emacs — это расширяемый текстовый редактор, который можно кастомизировать под свои нужды при помощи языка Emacs Lisp (Elisp).

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

Количество комбинаций клавиш и далеко не простой интерфейс могут показаться сложной частью для освоения новичком.

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

Буфер - это объект, содержащий записанный текст.

Окно - область прямоугольной формы, отображающая один из буферов.

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

Да, это возможно.

5. Какие буферы создаются по умолчанию при запуске emacs?

Messages, GNU Emacs, scratch.

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

Ctrl+C, а потом | и Ctrl+C, Ctrl + |.

7. Как поделить текущее окно на две части?

С помощью комбинаций клавиш Ctrl + x, 2 (по горизонтали) или Ctrl + x, 3 (по вертикали).

8. В каком файле хранятся настройки редактора emacs?

Настройки редактор emacs хранятся в файле .emacs, расположенным в директории пользователя.

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

Она выполняет функцию “стереть”.

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

Однозначно emacs, поскольку в vi неудобное переключение режимов и не имеем простого интерфейса.

## **6 Выводы**

В течение лабораторной работы я познакомилась с операционной системой Linux и получила практические навыки работы с редактором Emacs.



## Список литературы

1. Что такое Emacs? [Электронный ресурс]. URL: <https://ru.theastrologypage.com/emacs>.