

Отчет по лабораторной работе №11

Текстовый редактор emacs

Чигладзе Майя Владиславовна

Содержание

1	Цель работы	5
2	Порядок выполнения лабораторной работы	6
2.1	Задание 1. Открыла emacs	6
2.2	Задание 2. Создала файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).	6
2.3	Задание 3. Набрала текст	7
2.4	Задание 4. Сохранила файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s)	8
2.5	Задание 5. Прodelала с текстом стандартные процедуры редактирования.	8
2.6	Задание 6. Научилась использовать команды по перемещению курсора.	14
2.7	Задание 7. Управление буферами	16
2.8	Задание 8. Управление окнами.	18
2.9	Задание 9. Режим поиска	19
3	Ответы на контрольные вопросы	22
4	Выводы	24
	Список литературы	25

Список иллюстраций

2.1	Открытие	6
2.2	Создание	7
2.3	Набор текста	7
2.4	Сохранение	8
2.5	Вырез	9
2.6	Вставка	10
2.7	Выделение области	11
2.8	Копия области	11
2.9	Вставка в конец	12
2.10	Выделение	13
2.11	Отмена	13
2.12	Перемещение курсора в начало	14
2.13	Перемещение курсора в конец	15
2.14	Перемещение курсора в начало	15
2.15	Перемещение в конец	16
2.16	Список активных буферов	16
2.17	Перемещение в окна	17
2.18	Закрытие окон	17
2.19	Перемещение между буферами	18
2.20	Деление фрейма	18
2.21	Новый буфер	19
2.22	Режим поиска С	19
2.23	Результат поиска С	20
2.24	Выход из поиска	20
2.25	Режим поиска и замены М	20
2.26	Режим поиска М	21

Список таблиц

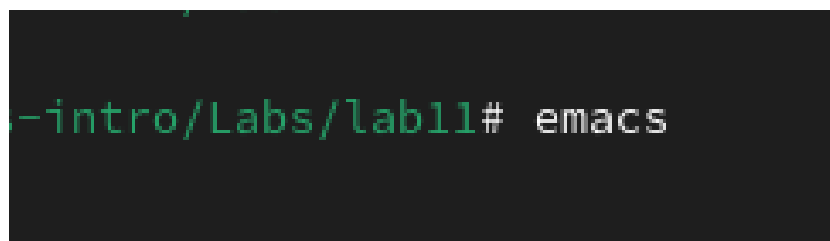
1 Цель работы

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

2 Порядок выполнения лабораторной работы

2.1 Задание 1. Открыла emacs

Результат (рис. 1).



```
-intro/Labs/lab11# emacs
```

Рис. 2.1: Открытие

2.2 Задание 2. Создала файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).

Результат (рис. 2).

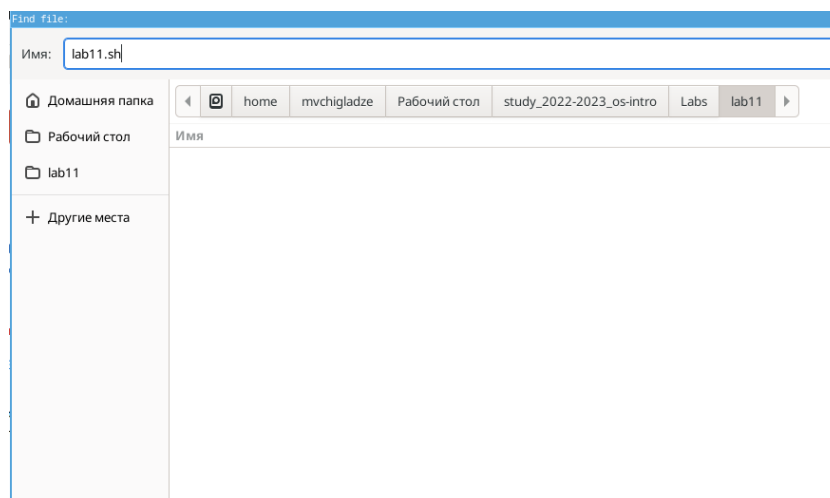


Рис. 2.2: Создание

2.3 Задание 3. Набрала текст

Результат (рис. 3).

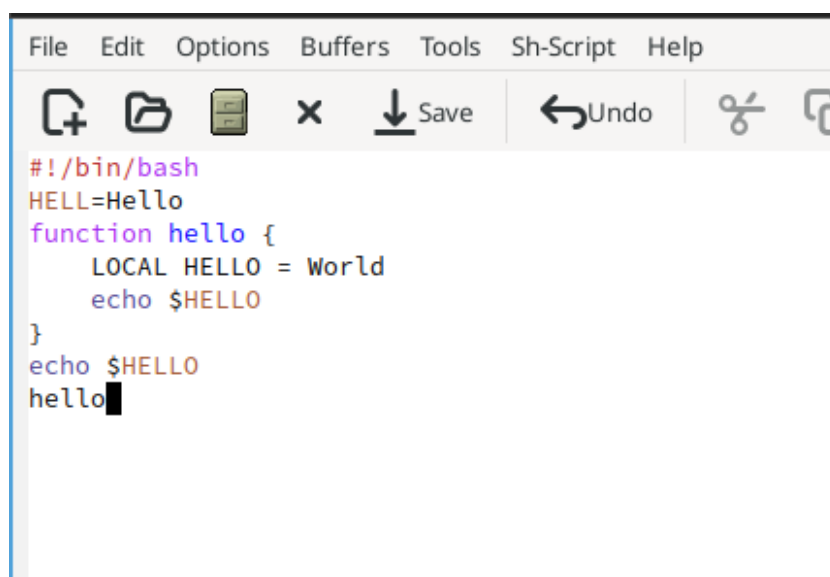
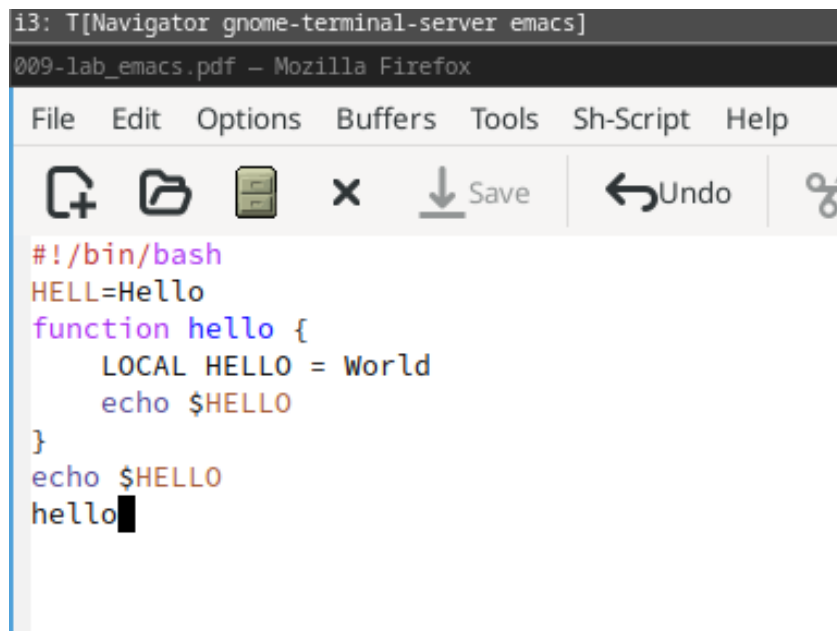


Рис. 2.3: Набор текста

2.4 Задание 4. Сохранила файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s)

Результат (рис. 4).



```
i3: T[Navigator gnome-terminal-server emacs]
009-lab_emacs.pdf - Mozilla Firefox
File Edit Options Buffers Tools Sh-Script Help
[Icons: New, Open, Save, Close, Save, Undo, Cut]
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO = World
    echo $HELLO
}
echo $HELLO
hello
```

Рис. 2.4: Сохранение

2.5 Задание 5. Проделала с текстом стандартные процедуры редактирования.

1. Вырезала одной командой целую строку (C-k) (рис. 5).

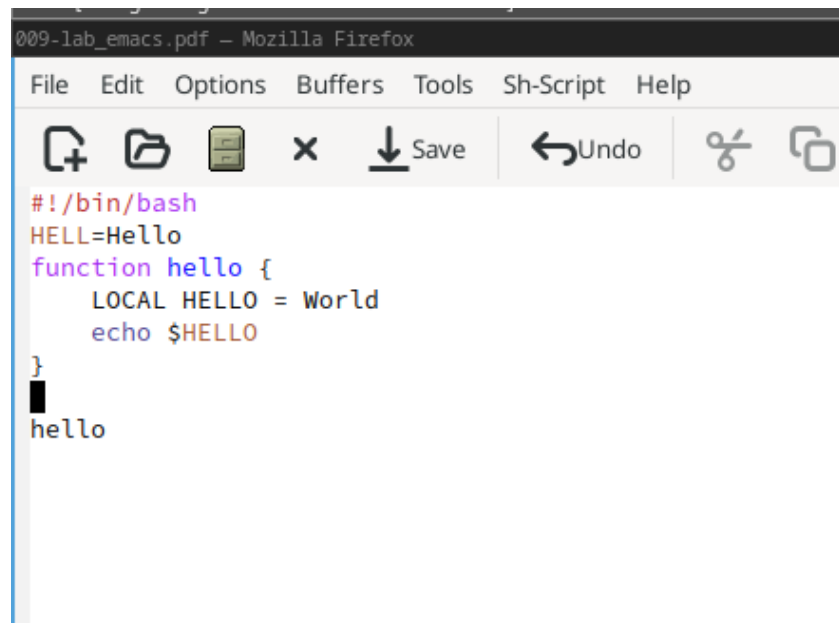


Рис. 2.5: Вырез

2. Вставила эту строку в конец файла (С-у)(рис. 6).

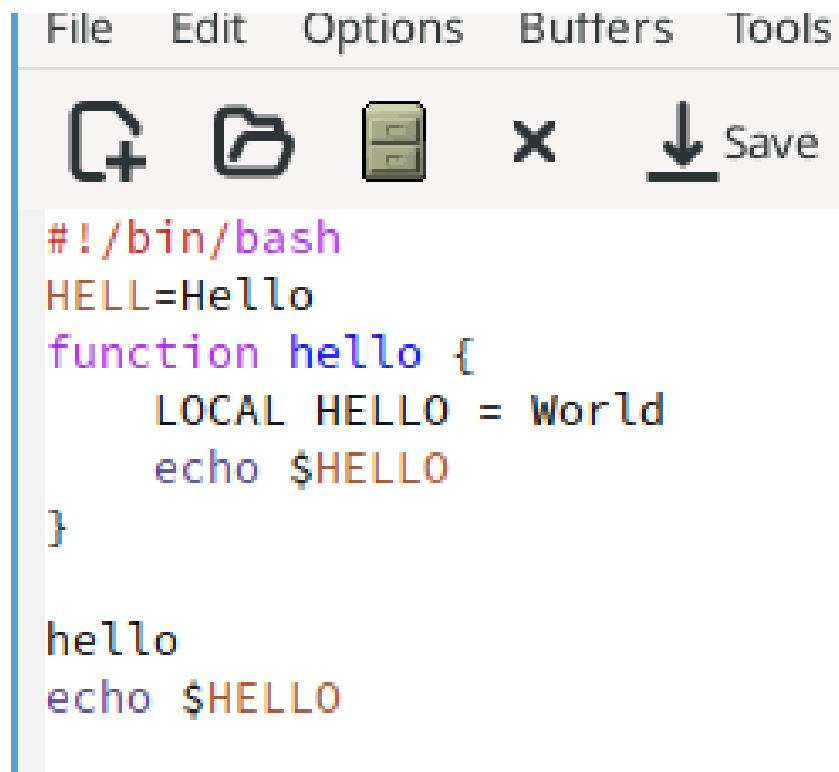
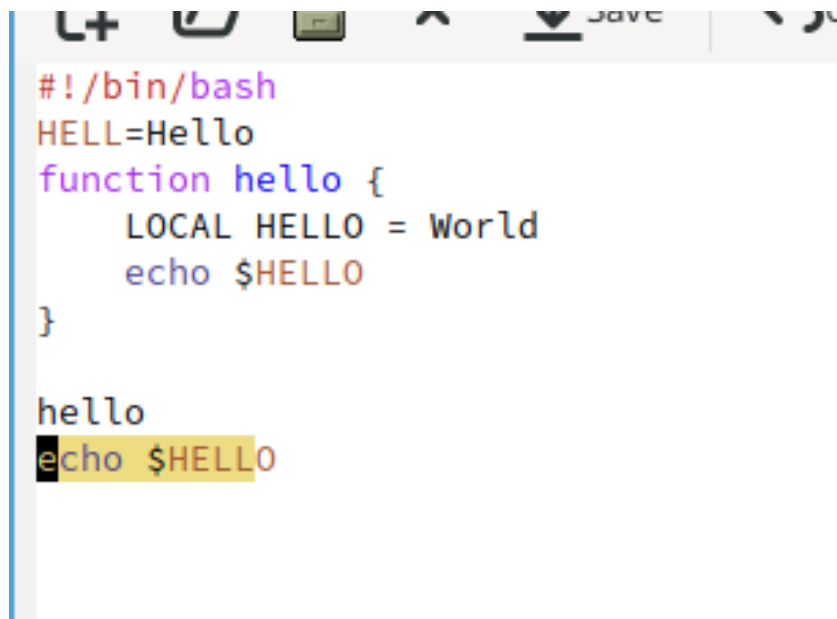


Рис. 2.6: Вставка

3. Выделила область текста (C-space) (рис. 7).



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

hello
echo $HELLO
```

Рис. 2.7: Выделение области

4. Скопировала область в буфер обмена (M-w)(рис. 8).

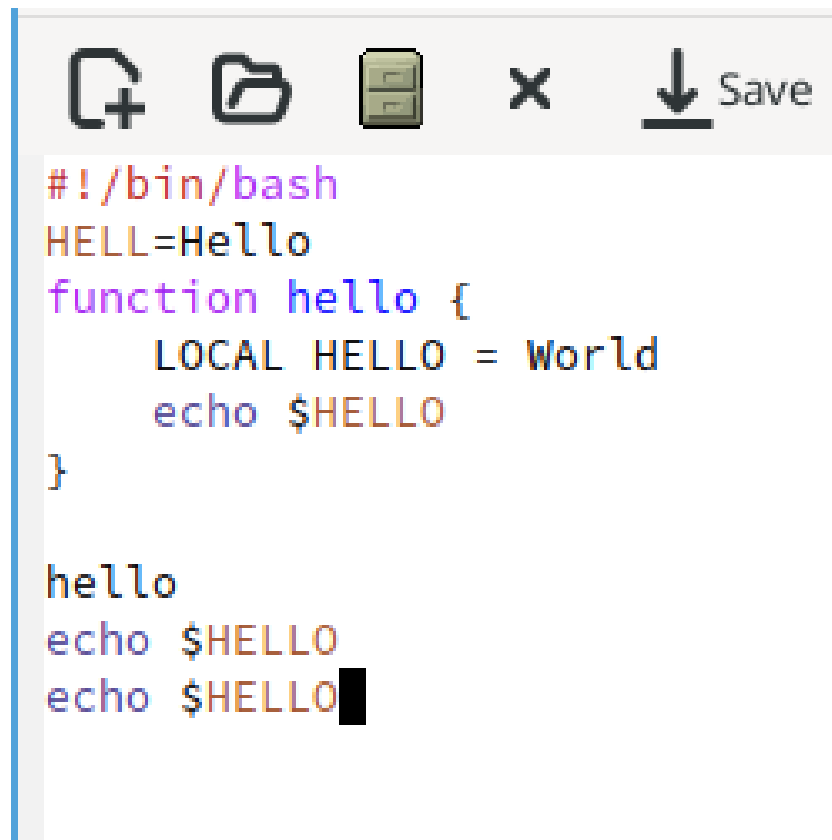


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO = World
    echo $HELLO
}

hello
echo $HELLO
```

Рис. 2.8: Копия области

5. Вставила область в конец файла (рис. 9).

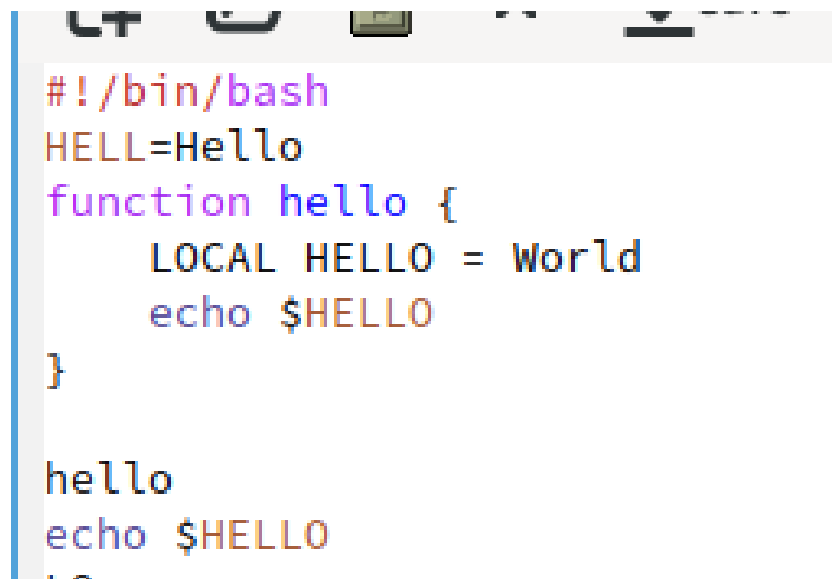


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO = World
    echo $HELLO
}

hello
echo $HELLO
echo $HELLO
```

Рис. 2.9: Вставка в конец

6. Вновь выделила эту область и на этот раз вырезала её (C-w) (рис. 10).

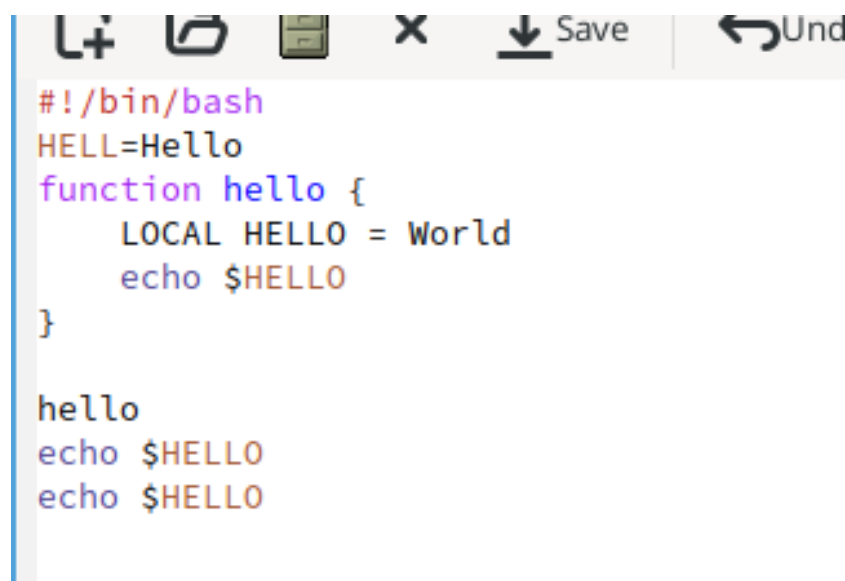


```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO = World
    echo $HELLO
}

hello
echo $HELLO
```

Рис. 2.10: Выделение

7. Отменила последнее действие (C-/) (рис. 11).



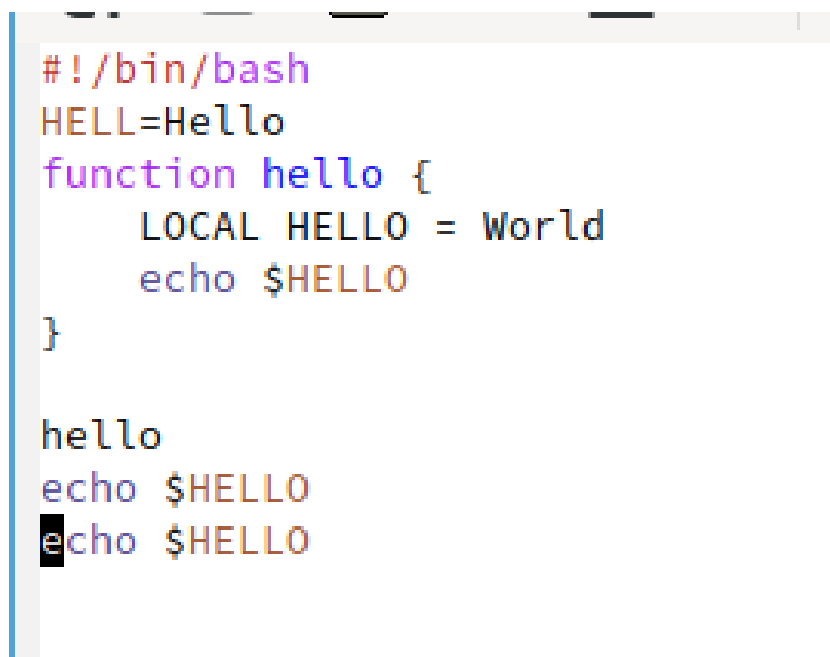
```
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO = World
    echo $HELLO
}

hello
echo $HELLO
echo $HELLO
```

Рис. 2.11: Отмена

2.6 Задание 6. Научилась использовать команды по перемещению курсора.

1. Переместила курсор в начало строки (C-a) (рис. 12).



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

hello
echo $HELLO
echo $HELLO
```

Рис. 2.12: Перемещение курсора в начало

2. Переместила курсор в конец строки (C-e) (рис. 13).

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

hello
echo $HELLO
echo $HELLO
```

Рис. 2.13: Перемещение курсора в конец

3. Переместила курсор в начало буфера (M-<) (рис. 14).

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

hello
echo $HELLO
echo $HELLO
```

Рис. 2.14: Перемещение курсора в начало

4. Переместила курсор в конец буфера (M->)(рис. 15).

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

hello
echo $HELLO
echo $HELLO
```

Рис. 2.15: Перемещение в конец

2.7 Задание 7. Управление буферами

1. Вывела список активных буферов на экран (C-x C-b)(рис. 16).

CRM	Buffer	Size	Mode	File
	* lab11.sh	112	Shell-script[sh]	/home/mvchigladze/Рабочи
%	*GNU Emacs*	734	Fundamental	
	scratch	145	Lisp Interaction	
%*	*Messages*	1168	Messages	
%*	*Async-native-compile-log*	165	Fundamental	

Рис. 2.16: Список активных буферов

2. Переместила во вновь открытое окно (C-x) о со списком открытых буферов и переключилась на другой буфер (рис. 17).

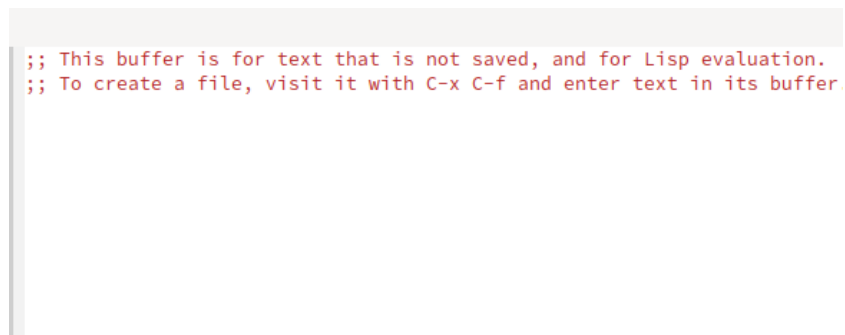


Рис. 2.17: Перемещение в окна

3. Закрывает это окно (C-x 0) (рис. 18).

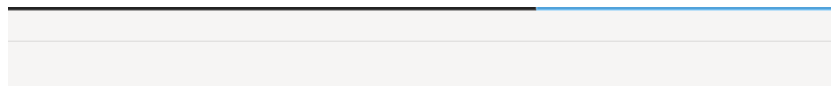


Рис. 2.18: Закрывание окон

4. Теперь вновь переключилась между буферами, но уже без вывода их списка на экран (C-x b) (рис. 19).

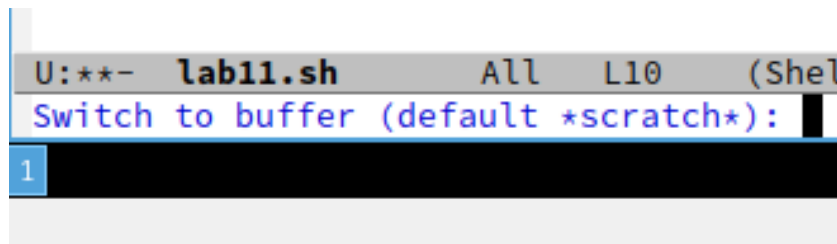


Рис. 2.19: Перемещение между буферами

2.8 Задание 8. Управление окнами.

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. 20).

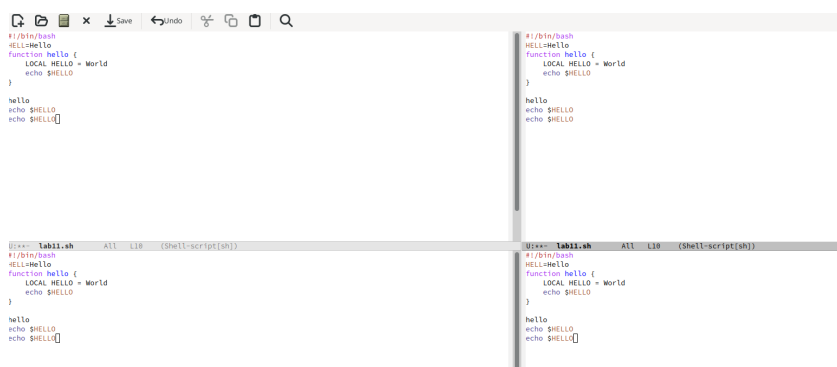


Рис. 2.20: Деление фрейма

2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста (рис. 21).

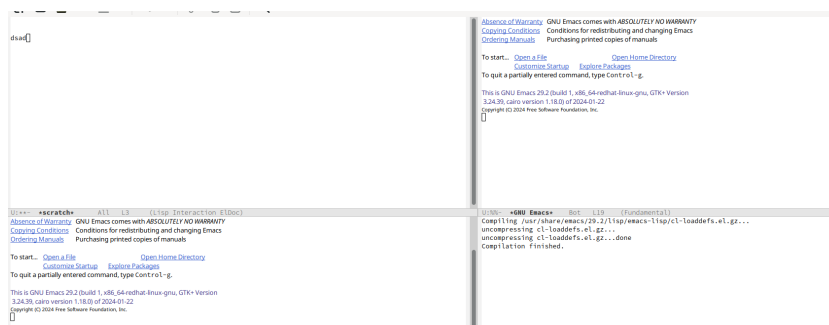


Рис. 2.21: Новый буфер

2.9 Задание 9. Режим поиска

1. Переключилась в режим поиска (C-s) и нашла несколько слов, присутствующих в тексте рис. 22).

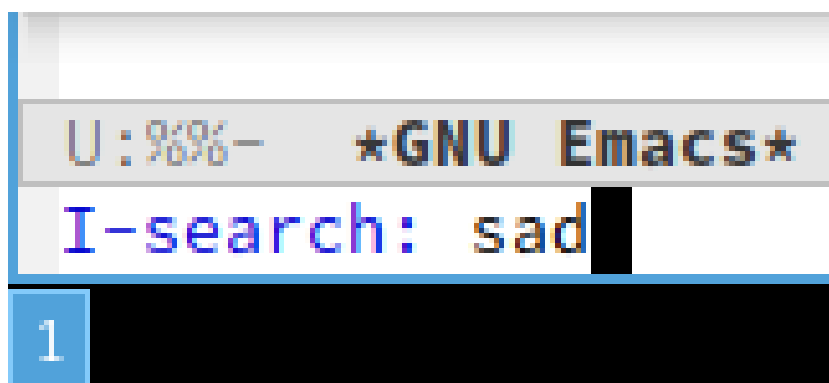


Рис. 2.22: Режим поиска C

2. Переключилась между результатами поиска, нажимая C-s (рис. 23).

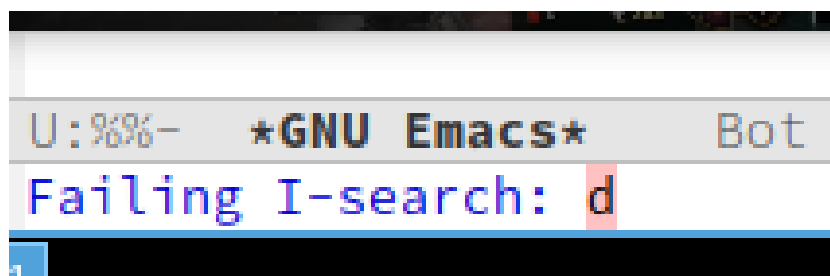


Рис. 2.23: Результат поиска C

3. Вышла из режима поиска, нажав C-g (рис. 24).

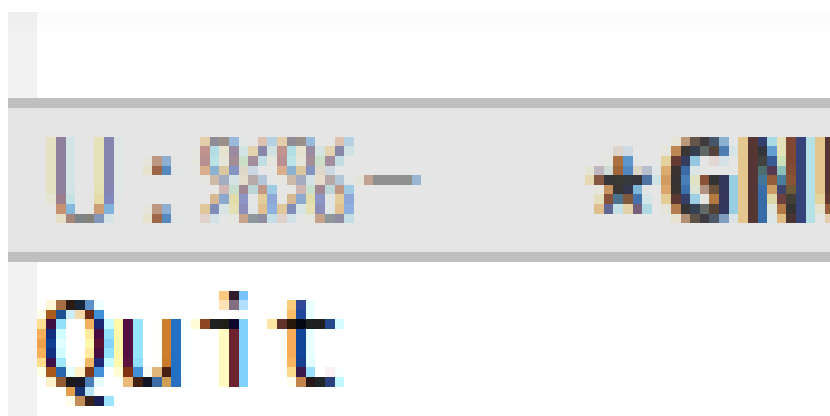


Рис. 2.24: Выход из поиска

4. Перешла в режим поиска и замены (M-%), ввела текст, который следует найти и заменить, нажали Enter, затем ввела текст для замены. После того как будут подсвечены результаты поиска, нажали ! для подтверждения замены. (рис. 25).

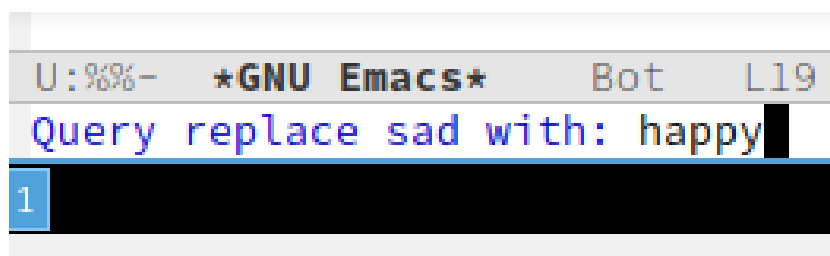


Рис. 2.25: Режим поиска и замены M

5. Испробовала другой режим поиска, нажав M-s o. Он отличается тем, что ищет по всем буферам и окнам. (рис. 26).

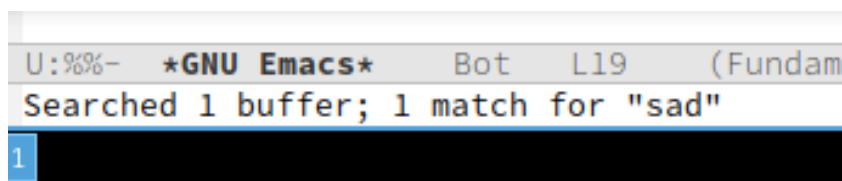


Рис. 2.26: Режим поиска M

3 Ответы на контрольные вопросы

1. Emacs - это текстовый редактор, написанный на языке Emacs Lisp, который предлагает многофункциональность и гибкость. Он имеет встроенный интерпретатор, поддерживает множество языков программирования и может быть расширен с помощью пользовательских пакетов.
2. Особенности редактора Emacs, которые могут сделать его сложным: обилие горячих клавиш, отсутствие интерфейса с графическим меню, необходимость знания языка Emacs Lisp для глубокой настройки.
3. Буфер в терминологии Emacs – это файл, с которым в данный момент работает редактор. Окно – это область на экране, в которой отображается содержимое буфера.
4. Да, можно открыть более 10 буферов в одном окне, используя различные режимы просмотра.
5. При запуске Emacs по умолчанию создаются буферы для инициализации основных систем редактора, включая режим загрузки файла, окно сообщений и командную строку.
6. Для ввода комбинации C-с | (Control-с вертикальное разделение) и C-с C- (Control-с удаление) необходимо нажать клавиши Ctrl, с, и затем соответствующую клавишу (вертикальное разделение или удаление).
7. Чтобы разделить текущее окно Emacs на две части, используйте комбинацию C-x 2 (Control-x два), которая вставляет горизонтальный разделитель в текущую позицию курсора.
8. Настройки редактора Emacs хранятся в файле .emacs.d/init.el в домашнем

каталоге пользователя.

9. Клавиша ‘ обычно используется для изменения регистра символов и не может быть переназначена.
10. Я предпочитаю Emacs из-за его гибкости и возможности настройки под свои нужды. Однако, если вы только начинаете изучать редактирование текста, Emacs может показаться сложным из-за обилия горячих клавиш и отсутствия графического меню. В этом случае, редактор vi может быть более подходящим вариантом.

4 Выводы

В ходе лабораторной работы, я познакомилась с операционной системой Linux. Получила практические навыки работы с редактором Emacs.

Список литературы