

Лабораторная работа №9

Текстовый редактор emacs

Захаренко Анастасия Викторовна

Содержание

Цель работы	5
Задание	6
Теоретическое введение	7
Выполнение лабораторной работы	8
Выводы	18
Список литературы	19

Список иллюстраций

0.1	выполнение1	8
0.2	выполнение2	9
0.3	lab07.sh	10
0.4	редактирование1	10
0.5	редактирование2	11
0.6	управление буферами1	11
0.7	управление буферами2	12
0.8	управление буферами3	13
0.9	управление окнами	13
0.10	поиск1	14
0.11	поиск2	14
0.12	поиск3	15
0.13	поиск4	15
0.14	поиск5	16
0.15	поиск6	17

Список таблиц

Цель работы

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

Задание

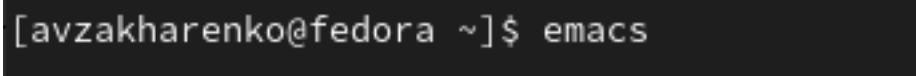
1. Ознакомиться с теоретическим материалом.
2. Ознакомиться с редактором etascs.
3. Выполнить упражнения.

Теоретическое введение

Определение 1. Буфер — объект, представляющий какой-либо текст. Буфер может содержать что угодно, например, результаты компиляции программы или встроенные подсказки. Практически всё взаимодействие с пользователем, в том числе интерактивное, происходит посредством буферов. Определение 2. Фрейм соответствует окну в обычном понимании этого слова. Каждый фрейм содержит область вывода и одно или несколько окон Emacs. Определение 3. Окно — прямоугольная область фрейма, отображающая один из буферов. Каждое окно имеет свою строку состояния, в которой выводится следующая информация: название буфера, его основной режим, изменялся ли текст буфера и как далеко вниз по буферу расположен курсор. Каждый буфер находится только в одном из возможных основных режимов. Существующие основные режимы включают режим Fundamental (наименее специализированный), режим Text, режим Lisp, режим C, режим Texinfo и другие. Под второстепенными режимами понимается список режимов, которые включены в данный момент в буфере выбранного окна. Определение 4. Область вывода — одна или несколько строк внизу фрейма, в которой Emacs выводит различные сообщения, а также запрашивает подтверждения и дополнительную информацию от пользователя. Определение 5. Минibuфер используется для ввода дополнительной информации и всегда отображается в области вывода. Определение 6. Точка вставки — место вставки (удаления) данных в буфере.

Выполнение лабораторной работы

1. Открываем emacs.



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

Рис. 0.1: выполнение1

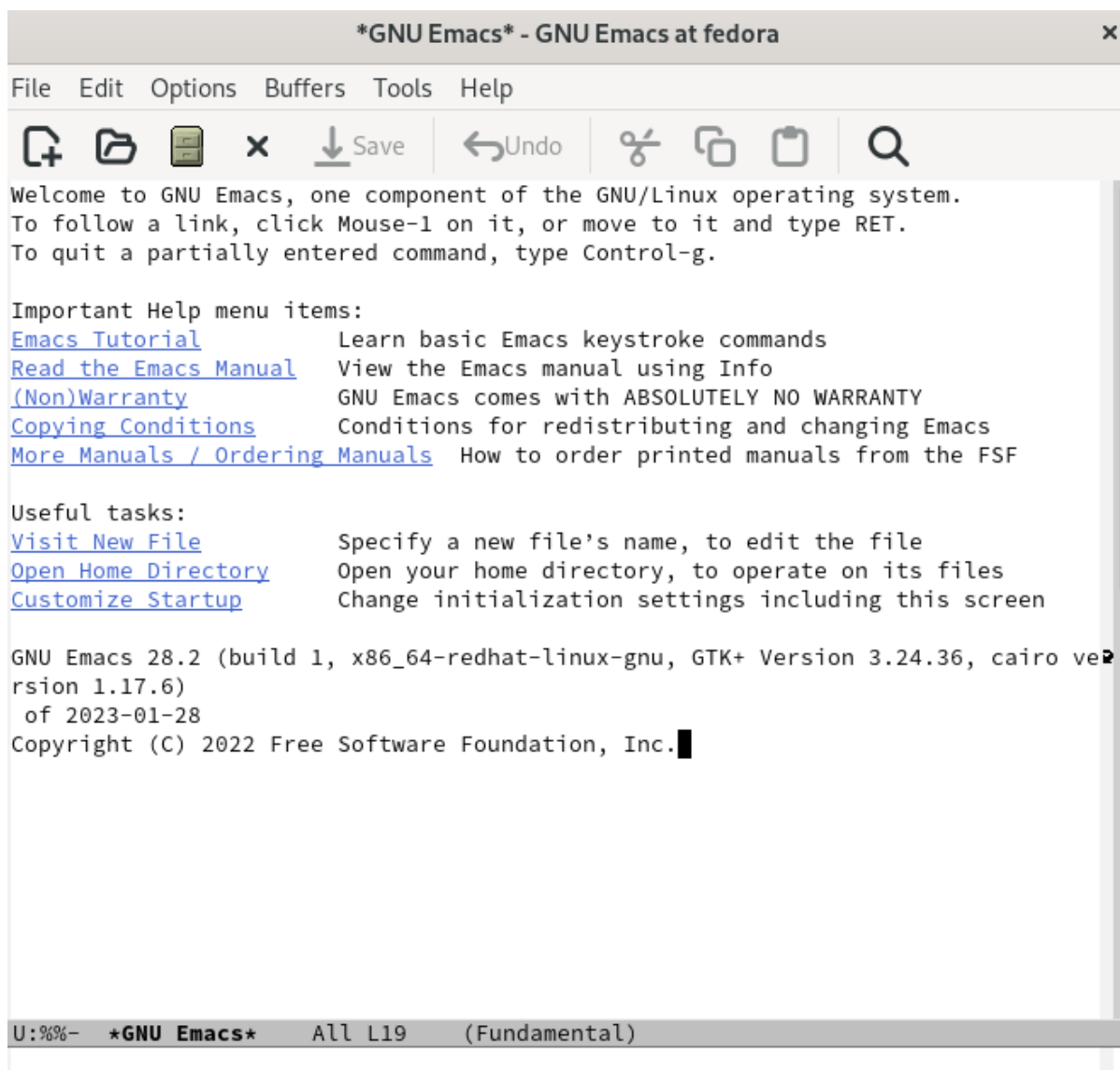


Рис. 0.2: выполнение2

2. Создаем файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).
3. Набираем текст: 1 #!/bin/bash 2 HELL=Hello 3 function hello { 4 LOCAL
HELLO=World 5 echo \$HELLO 6 } 7 echo \$HELLO 8 hello
4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).

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

Рис. 0.3: lab07.sh

5. Прodelать с текстом стандартные процедуры редактирования, каждое действие долж- но осуществляться комбинацией клавиш. 5.1. Вырезать одной командой целую строку (C-k). 5.2. Вставить эту строку в конец файла (C-y).

```
#!/bin/bash

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

Рис. 0.4: редактирование1

- 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-/).

```
#!/bin/bash

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

Рис. 0.5: редактирование2

6. Управление буферами. 6.1. Вывести список активных буферов на экран (C-x C-b)

-:***- lab07.sh		All L13	(Shell-script[bash])
CRM Buffer		Size	Mode File
□ * lab07.sh		151	Shell-script[ba... ~/lab07.sh
% *GNU Emacs*		904	Fundamental
scratch		145	Lisp Interaction
%* *Messages*		626	Messages

Рис. 0.6: управление буферами1

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

```
#!/bin/bash

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

~:***- lab07.sh All L10

```
#!/bin/bash

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

Рис. 0.7: управление буферами2

6.3. Закройте это окно (C-x 0). 6.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b).

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

Рис. 0.8: управление буферами3

7. Управление окнами. 7.1. 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)

```
#!/bin/bash

function hello {
    LOCAL HELL=World
    echo $HELLO
}

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

lab07.sh All L10 (Shell-s

```
#!/bin/bash


function hello {
    LOCAL HELL=World
    echo $HELLO
}

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

lab07.sh All L10 (Shell-s

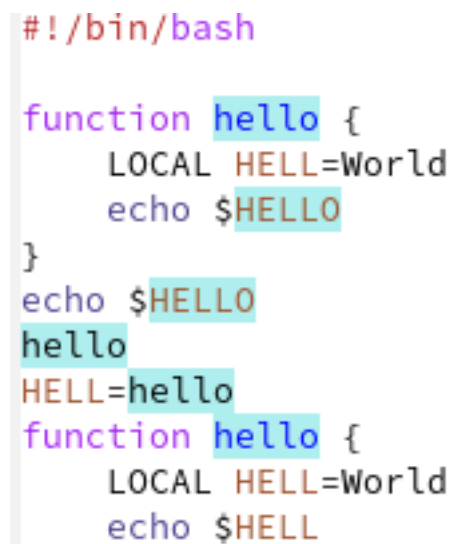
Рис. 0.9: управление окнами

8. Режим поиска 8.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте.



```
Failing I-search: hello
```

Рис. 0.10: поиск1



```
#!/bin/bash

function hello {
    LOCAL HELL=World
    echo $HELLO
}

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

Рис. 0.11: поиск2

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

```

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

```

Рис. 0.12: поиск3

```

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

```

Рис. 0.13: поиск4

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

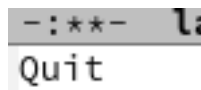


Рис. 0.14: поиск5

8.4. Испробуйте другой режим поиска, нажав M-s o.


```
#!/bin/bash

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

```
-:***- lab07.sh All L12 (Shell-scr
6 matches for "hello" in buffer: lab07.sh
    :#!/bin/bash
    :
3: function hello {
    :     LOCAL HELL=World
5:     echo $HELLO
    :}
7: echo $HELLO
8: hello
9: HELL=hello
10: function hello {
    :     LOCAL HELL=World
    :     echo $HELL
```

Рис. 0.15: поиск6

Выводы

Я познакомилась с операционной системой Linux. Получила практические навыки работы с редактором Emacs

Список литературы