

Лабораторная работы №9 “Текстовой редактор emacs”

Тулёуов Мадн

RUDN University, Moscow, Russian Federation

Цель работы:

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

1. Открыть emacs.
2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).
3. Набрать, приведенный в документе, текст.
4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).

5. 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-/).

7. Управление буферами.

- 7.1 Вывести список активных буферов на экран (C-x C-b).
- 7.2 Переместиться во вновь открытое окно (C-x) о со списком открытых буферов и переключиться на другой буфер.
- 7.3 Закрыть это окно (C-x 0).
- 7.4 Теперь вновь переключиться между буферами, но уже без вывода их списка на экран (C-x b).

8. Управление окнами.

- 8.1 Поделить фрейм на 4 части: разделить фрейм на два окна по вертикали (С-х 3), а затем каждое из этих окон на две части по горизонтали (С-х 2)
- 8.2 В каждом из четырёх созданных окон открыть новый буфер (файл) и ввести несколько строк текста.

9. Режим поиска

- 9.1 Переключиться в режим поиска (C-s) и найти несколько слов, присутствующих в тексте.
- 9.2 Переключиться между результатами поиска, нажимая C-s.
- 9.3 Выйти из режима поиска, нажав C-g.
- 9.4 Перейти в режим поиска и замены (M-%), ввести текст, который следует найти и заменить, нажмите Enter , затем ввести текст для замены. После того как будут подсвечены результаты поиска, нажать ! для подтверждения замены.

5. Испробовать другой режим поиска, нажав М-с о.
Объяснить, чем он отличается от обычного режима?

Открыл emacs. (рис. 1)

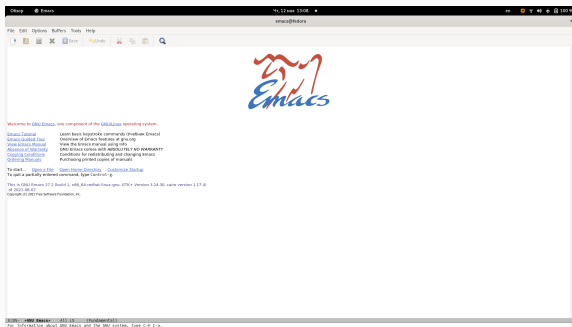


Figure 1: Emacs

Ход работы

Создал файл lab07.sh с помощью комбинации Ctrl-x
Ctrl-f. Затем я ввел, приведенный в документе, текст.(рис. 2)

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


Вырезал одной командой целую строку.(рис. 4)

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

hello
```

Figure 4: Вырезание строки

Вставил эту строку в конец файла.(рис. 5)



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

hello
echo $HELLO
```

Figure 5: вставка строки

Ход работы

Выделил область текста, скопировал область в буфер обмена и вставил его в конец файла. (рис. 6)

A screenshot of a terminal window with a light gray background. The terminal shows a shell script being edited or executed. The script defines a function named 'hello' that sets a local variable 'HELLO' to 'World' and echoes the value of 'HELLO'. Below the function definition, the word 'hello' is entered on a new line, followed by two lines of 'echo \$HELLO'. The cursor is at the end of the second 'echo' line.

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

hello
echo $HELLO
echo $HELLO
```

Вновь выделил эту область и вырезал ее.(рис. 7)

A screenshot of a terminal window. The window has a title bar with standard Linux window controls (minimize, maximize, close) and a toolbar with icons for file operations (new, open, save, etc.). The terminal content shows a bash script being executed. The script defines a function named 'hello' that sets a local variable 'HELLO' to 'World' and echoes it. After the function definition, the 'hello' function is called, and then 'echo \$HELLO' is executed, which prints 'World'.

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

hello
echo $HELLO
```

Отменил последнее действие.(рис. 8)

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

hello
echo $HELLO
echo $HELLO
```

Figure 8: Отмена последнего действия

Переместил курсор в начало строки (C-a).(рис. 9)

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

hello
echo $HELLO
echo $HELLO
```

Figure 9: Перемещение курсора в начало строки

Переместил курсор в конец строки. (рис. 10)

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

hello
echo $HELLO
echo $HELLO
```

Figure 10: Перемещение курсора в конец строки

Переместил курсор в начало буфера, а затем в конец буфера.

Вывел список активных буферов на экран.(рис. 11)

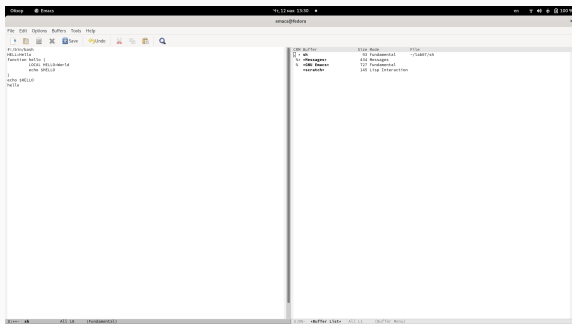


Figure 11: Активные буферы

Закрыл это окно. (рис. 13)

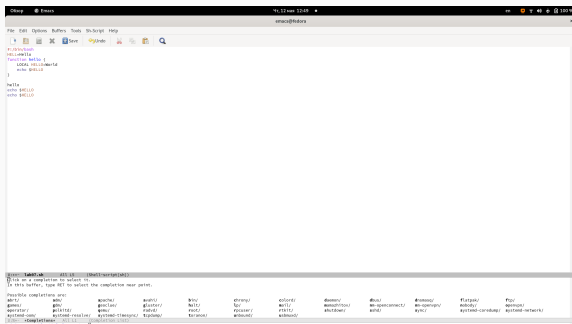


Figure 13: Заккрытие окна

Переключился на другой буфер, но уже без вывода их списка на экран.(рис. 14)

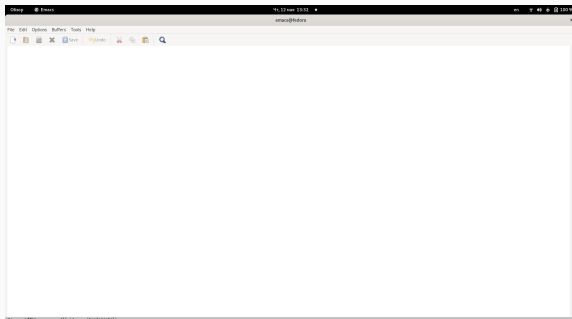


Figure 14: Переключение на другой буфер без вывода списка

Поделил фрейм на 4 части: разделил фрейм на два окна по вертикали, а затем каждое из этих окон на две части по горизонтали.(рис. 15)

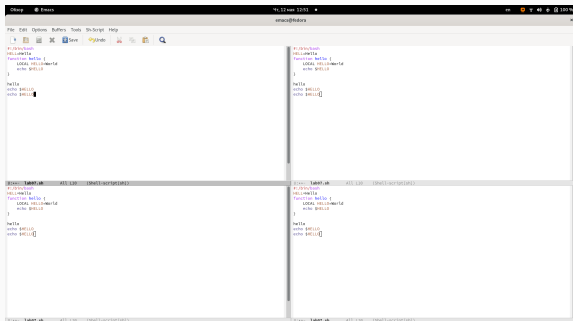


Figure 15: Деление фрейма на 4 части

В каждом из четырёх созданных окон открыл новый буфер (файл) и ввел случайные буквы.(рис. 16)

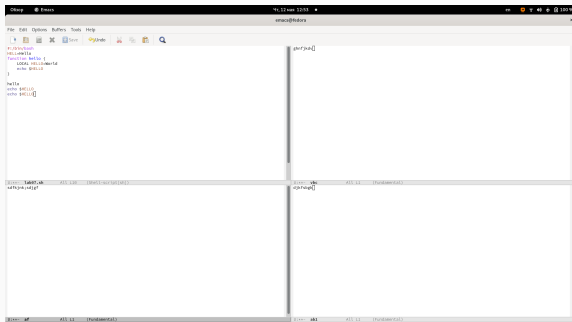


Figure 16: Введение текста в 4 окнах

Переключился в режим поиска и нашел слово *hello*. (рис. 17)

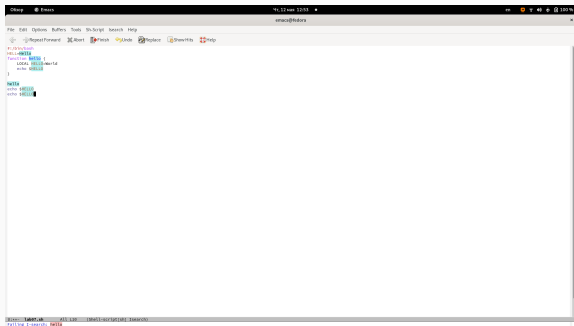


Figure 17: Поиск слова

Переключился между результатами поиска. (рис. 18)

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

hello
echo $HELLO
echo $HELLO
```

Figure 18: Переключение между результатами поиска

Испробовал другой режим поиска, нажав M-s o. (рис. 19)

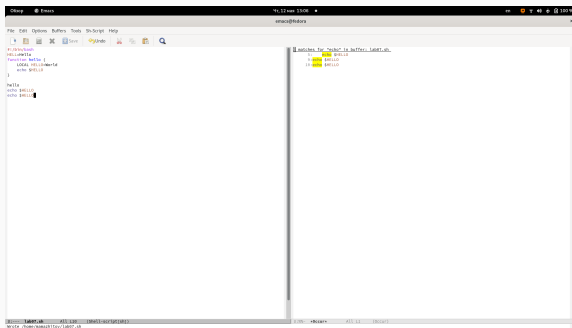


Figure 19: 2 способ поиска слов

Мы получили базовые навыки использования etacs.