

Лабораторная работа №10

Журавлев Георгий Иванович

Цель работы

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

1. Открыл emacs.



Рис. 1: команда для открытия редактора

2. Создал файл lab07.sh с помощью комбинации (C-x C-f).



Рис. 2: создание файла

3. Набрал текст:
4. Сохранил файл с помощью комбинации (C-x C-s).(wrote /home/g_zhuravlev/lab07.sh)
5. Прodelал с текстом стандартные процедуры редактирования, каждое действие осуществлялось комбинацией клавиш:
 - 5.1. Вырезал одной командой целую строку (C-k).
 - 5.2. Вставил эту строку в конец файла (C-y).
 - 5.3. Выделил область текста (C-space).
 - 5.4. Скопировал область в буфер обмена (M-w).
 - 5.5. Вставил область в конец файла.
 - 5.6. Ещё раз выделил эту область и на этот раз удалил её в буфер обмена(C-w).

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

```
U:--- lab07.sh All L8 (Shell-script[sh])
Wrote /home/g zhuravlev/lab07.sh
```

Рис. 3: набранный текст

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

```
U:--- lab07.sh All L8 (Shell-script[sh])
Wrote /home/g zhuravlev/lab07.sh
```

Рис. 4: сохранение файла

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

Рис. 5: до удаления строки

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

Рис. 6: после удаления строки

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

hello
echo $HELLO
```

Рис. 7: вставка строки в конец файла

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

hello
echo $HELLO
```

Рис. 8: выделение и копирование

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

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

hello
echo $HELLO
```

Рис. 9: вставка

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

hello
echo $HELLO
```

Рис. 10: удаление области

- 5.7. Отменил последнее действие (C-/).
6. Использовал некоторые команды по перемещению курсора:
 - 6.1. Переместил курсор в начало строки (C-a).
 - 6.2. Переместил курсор в конец строки (C-e).
 - 6.3. Переместил курсор в начало буфера (M-<).
 - 6.4. Переместил курсор в конец буфера (M->).
7. Управление буферами.
 - 7.1. Вывел список активных буферов на экран (C-x C-b).
 - 7.2. Переместился во вновь открытое окно (C-x) о со списком открытых буферов и переключился на другой.
 - 7.3. Закрыл это окно (C-x 0).
 - 7.4. Вновь переключился между буферами, но без вывода их списка на экран (C-x b).
8. Управление окнами.
 - 8.1. Поделил фрейм на 4 части: разделил 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 открыл новый буфер (файл) и ввёл несколько строк текста.

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

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

hello
echo $HELLO
```

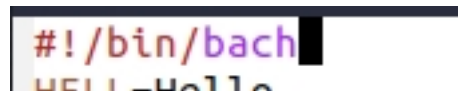
Рис. 11: отмена действия

```
#!/bin/bash
HELL=Hello
```

Рис. 12: до перемещения

```
#!/bin/bash
HELL=Hello
```

Рис. 13: после перемещения



```
#!/bin/bash
HELL=Hello
```

Рис. 14: результат перемещения



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

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

hello
echo $HELLO
```

Рис. 15: результат перемещения

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

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

hello
echo $HELLO
```

Рис. 16: результат перемещения

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

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

hello
echo $HELLO
```

U:***	lab07.sh	All	L1	(Shell-script[sh])
CRM	Buffer	Size	Mode	File
.	* lab07.sh	195	Shell-script[sh]	~/lab07.sh
%	*GNU Emacs*	705	Fundamental	
%	*scratch*	145	Lisp Interaction	
%*	*Messages*	1528	Messages	

Рис. 17: активные буферы



Рис. 18: иной буфер

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

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

echo $HELLO
```

Рис. 19: результат закрытия окна

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

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

echo $HELLO
```

U:***- lab07.sh All L17 (Shell-script[sh])
Switch to buffer (default *GNU Emacs*):

Рис. 20: до переключения

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

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

echo $HELLO
```

```
U:**- lab07.sh All L17 (Shell-script[sh])
user-error: Marker stack is empty
user-error: File /home/g_zhuravlev/TAGS does not exist [2 times]
C-. is undefined
Mark set [4 times]
Auto-saving...done
Use +,-,0 for further adjustment [15 times]
zsh: command not found: hello

Mark set
Use +,-,0 for further adjustment [19 times]
next-line: End of buffer
x is undefined
Mark set [4 times]
```

```
U:**- *Messages* Bot L46 (Messages)
```

Рис. 21: после переключения



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

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

echo $HELLO
```

U:*** lab07.sh All L17 (Shell-script[sh])

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

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

echo $HELLO
```

U:*** lab07.sh All L17 (Shell-script[sh])

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

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

echo $HELLO
```

U:*** lab07.sh All L17 (Shell-script[sh])

Рис. 22: результат деления окна

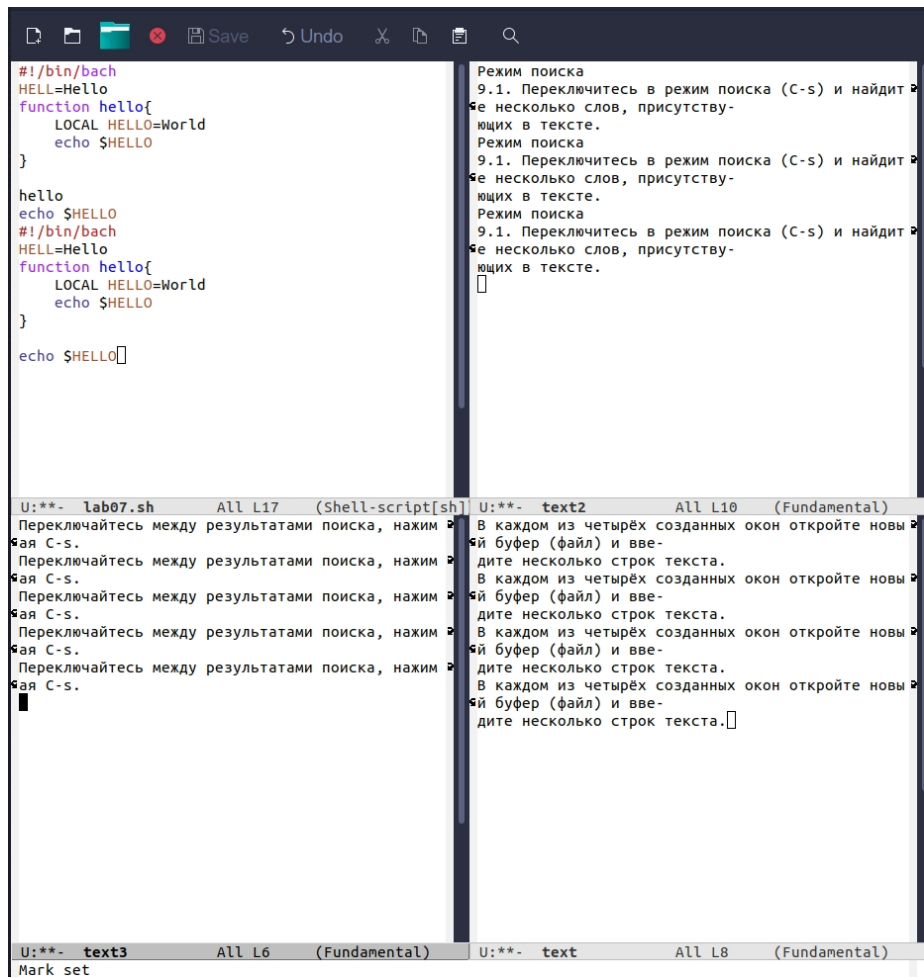


Рис. 23: ввод нового текста

9. Режим поиска.

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

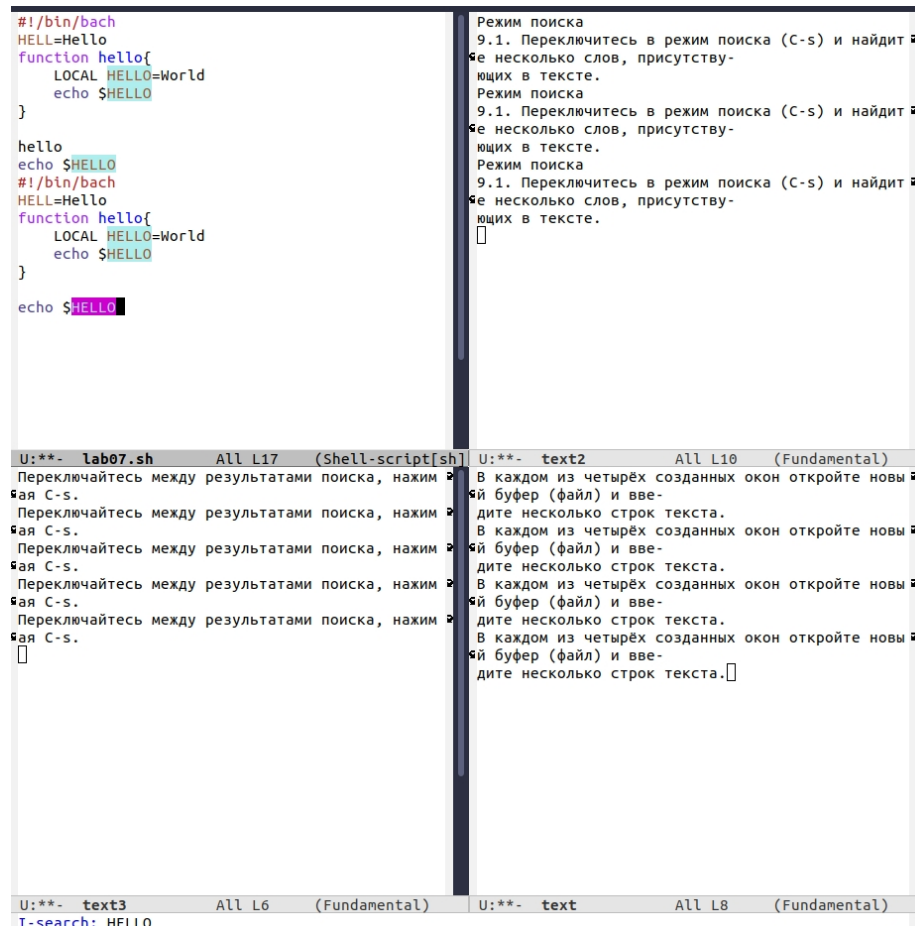


Рис. 24: результат поиска

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

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

9.4. Перешел в режим поиска и замены (M-%), ввел текст, который следует найти и заменить, нажал Enter.

9.5. Испробовал другой режим поиска, нажав M-s. Он отличается от обычного режима тем, что при поиске

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

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

echo $HELLO
```

```
U:**- lab07.sh All L4 (Shell-script[sh] Isee
Wrapped I-search: HELLO
```

Рис. 25: переключение между результатами

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

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

echo $HELLO
```

U:*** lab07.sh All L17 (Shell-script[sh])
Quit

Рис. 26: результат выхода

```
echo $HELLO LOCAL HELLO
```

```
U:**- lab07.sh All L17 (Shell-script[sh])  
Query replace: LOCAL HELLO
```

Рис. 27: замена текста

```
echo $HELLO LOCAL HELLO
```

```
J:**- lab07.sh All L17 (Shell-script[sh])  
Query replace LOCAL HELLO with: localgoodbye
```

Рис. 28: замена текста

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

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

echo $HELLO

LOCAL HELLO

```

U:**- lab07.sh All L20 (Shell-script[sh])

3 matches for "LOC" in buffer: lab07.sh

```
4:  LOCAL HELLO=World
13: LOCAL HELLO=World
19: LOCAL HELLO
```

U:%%- *Occur* All L4 (Occur)

Рис. 29: другой режим поиска

Вывод.

Благодаря этой лабораторной работе, я научился: исполнять некоторые команды в редакторе emacs; редактировать файлы в этом редакторе; пользоваться горячими клавишами редактора.

Контрольные вопросы:

1. Emacs - мощный экраный редактор текста, написанный на языке Elisp.
2. Программа - чрезвычайно сложна для освоения из-за своей многогранности.
3. Буфер – объект, состоящий из текста.
Окно – прямоугольная область с одним из буферов.
4. Да, в одном окне можно открыть больше 10 буферов.
5. После запуска emacs без каких-либо параметров в основном окне отображается буфер *scratch*, который содержит текст, введенный в командной строке.
6. Чтобы ввести C-c | я использую Ctrl+c и Shift+\ ; для C-c C-
|: Ctrl+c и Ctrl+Shift+\.
C-c C-|: Ctrl+c и Ctrl+Shift+|.
7. Поделить окно на две части можно двумя комбинациями клавиш:
C-x 3(вертикально) или C-x 2(горизонтально).
8. Настройки emacs хранятся в файле .emacs, который хранится в домашней директории пользователя.
9. Клавиша "enter" выполняет функцию перемещения курсора в открытом окне, и, её можно переназначить.
10. Редактор emacs более понятен для меня. Интерфейс и горячие клавиши, деление окна и т.п. Работать с ним удобнее.