

Лабораторная работа №11

Текстовый редактор emacs

Мухин Тимофей Владимирович, НБИбд-03-23

20.04.2024

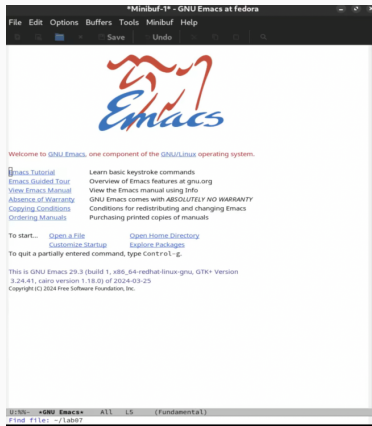
Российский университет дружбы народов, Москва, Россия

Цель работы

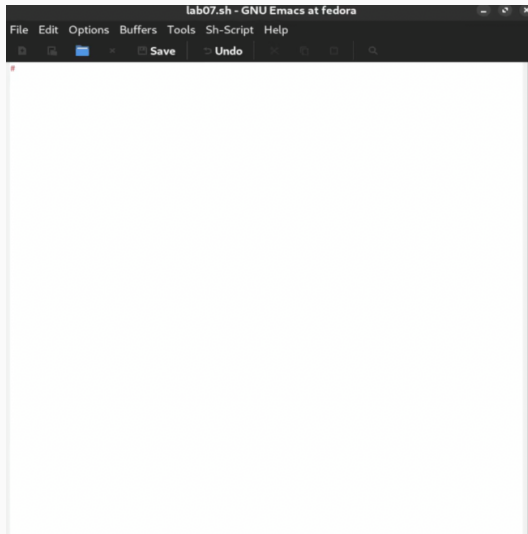
Познакомиться с операционной системой Linux. Получить практические навыки работы с редактором Emacs.

Выполнение работы.

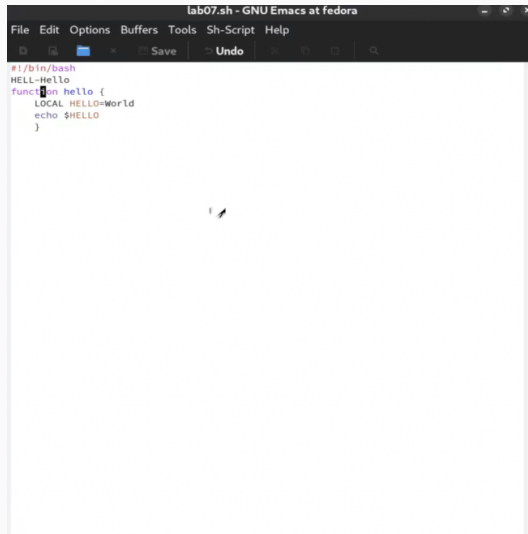
Открываем emacs



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



Набираем текст



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

Сохраняем файл


```
U:--- lab07.sh      All  L6      (Shell-script[sh])  
Wrote /home/seychik/lab07.sh
```


Проделаем с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш:

Вырезать одной командой целую строку (C-k)

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

Вставить эту строку в конец файла (С-у)



The screenshot shows a code editor window with a dark theme. The menu bar at the top includes icons for file operations and the text 'Save' and 'Undo'. The code in the editor is as follows:

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

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

The cursor is positioned at the end of the 'function hello {' line.

Выделить область текста (C-space)



The image shows a screenshot of a code editor window. The editor has a dark theme with a black background. The top toolbar includes icons for file operations and text editing, along with 'Save' and 'Undo' buttons. The code is written in a shell script syntax. The first line, `#!/bin/bash`, is highlighted in yellow. The second line, `HELL-Hello`, is highlighted in orange. The rest of the code is in white text. The code includes a function definition and a variable assignment.

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

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


Скопировать область в буфер обмена (M-w)

A screenshot of a terminal window with a dark theme. The window has a title bar with icons for file operations and a menu bar with 'Save' and 'Undo' options. The terminal content shows a bash script with a shebang, a function definition, and a function call. The text is color-coded: red for shebangs, purple for function names, and orange for variable names. A cursor is visible at the end of the last line of the script.

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

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

Вставить область в конец файла

A screenshot of a code editor window. The window has a dark title bar with icons for file operations and a menu bar with 'Save' and 'Undo' options. The code is written in a light-colored font on a white background. It starts with a shebang line, followed by a comment, then a function definition. The function 'hello' is currently open, with the cursor at the end of the opening brace.

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

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

Вновь выделить эту область и на этот раз вырезать её (C-w)

A screenshot of a code editor window. The window has a dark title bar with icons for file operations and a menu bar with 'Save' and 'Undo' options. The code is written in a light-colored font on a dark background. It shows a bash script starting with a shebang, followed by a function definition 'hello' that prints 'HELLO-Hello'. The cursor is positioned at the end of the function definition.

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

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

Отмените последнее действие (C-/)



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

function hello {
    LOCAL_HELLO=World
    echo $HELLO
}

function hello {
    #!/bin/bash
    HELL=Hello
```


Изучаем команды по перемещению курсора



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

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

The image shows a terminal window with a dark title bar containing icons for file operations and a search icon. The terminal content displays a shell script. The first line is a shebang, followed by a line with the text 'HELL-Hello'. There is a blank line, then a function definition. The cursor is positioned at the end of the line 'echo \$HELLO' within the function. The script continues with a closing brace, another function definition, and another 'HELL-Hello' line.

Управление буферами

```
-:***- lab07.sh All L7 (Shell-script[bash])
CRM Buffer Size Mode File
* lab07.sh 114 Shell-script[ba_ ~/lab07.sh
% *GNU Emacs* 718 Fundamental
*scratch* 145 Lisp Interaction
%* *Messages* 425 Messages
%* *Async-native-compile-l_ 165 Fundamental

>I
I:
5:
.
```

```
U:%%- *Buffer List* All L1 (Buffer Menu)
```

Перемещение в новое окно и переключение на другой буфер

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

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

~:*** lab07.sh All L7 (Shell-script[bash])

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

Переключение между буферами без вывода списка

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

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

~:*** lab07.sh All L7 (Shell-script[bash])

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

Управление окнами

Разделение фрейма на четыре части



The image shows a terminal window with a dark background and light-colored text. The window is divided into four panes by a vertical and a horizontal line. The top-left pane shows a script with a function definition. The top-right pane shows the same script but with the cursor on the line 'HELL-Hello'. The bottom-left pane shows the script with the function definition. The bottom-right pane shows the script with the function definition and the cursor on the line 'HELL-Hello'.

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

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

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

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

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

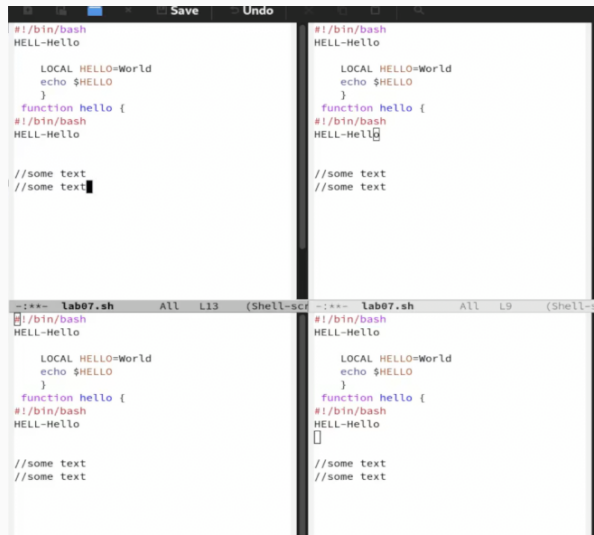
LOCAL HELLO=World
echo $HELLO
}
function hello {
#!/bin/bash
HELL-Hello
```

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

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

Выполнение работы

Открытие нового буфера в каждом из четырех окон



The image shows four terminal windows arranged in a 2x2 grid, each displaying the same script. The script contains a function 'hello' and a comment '//some text'. In each window, a new buffer is opened by pressing the 'Enter' key at the end of the line 'HELL-Hello'.

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

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

//some text
//some text
```

Режим поиска

Поиск и нахождение слов в тексте

<pre>#!/bin/bash HELL=Hello LOCAL HELLO=World echo \$HELLO } function hello { #!/bin/bash HELL=Hello</pre>	<pre>#!/bin/bash HELL=Hello LOCAL HELLO=World echo \$HELLO } function hello { #!/bin/bash HELL=Hello</pre>
<pre>--:100%-- lab07.sh All L2 (Shell-scf #!/bin/bash HELL=Hello LOCAL HELLO=World echo \$HELLO } function hello { #!/bin/bash HELL=Hello</pre>	<pre>--:100%-- lab07.sh All L8 (Shell-si #!/bin/bash HELL=Hello LOCAL HELLO=World echo \$HELLO } function hello { #!/bin/bash HELL=Hello </pre>

Режим поиска и замены

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

LOCAL HELLO=World
echo $HELLO
}

function hello {
#!/bin/bash
HELL=Hello

//some text
//some text
```

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

LOCAL HELLO=World
echo $HELLO
}

function hello {
#!/bin/bash
HELL=Hello

//some text
//some text
```

```
~:*** lab07.sh All L13 (Shell-scr
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}

function hello {
#!/bin/bash
HELL=Hello

//some text
//some text
```

```
~:*** lab07.sh All L9 (Shell-s
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}

function hello {
#!/bin/bash
HELL=Hello

//some text
//some text
```

Использование другого режима поиска

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

LOCAL HELLO=World
echo $HELLO
}

function hello \
#!/bin/bash
HELL=Hello

//some text
//some text1
```

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

LOCAL HELLO=World
echo $HELLO
}

function hello \
#!/bin/bash
HELL=Hello

//some text
//some text1
```

```
~:*** lab07.sh All L11 (Shell-scr -60- lab07.sh All L9 (Shell-scr
#!/bin/bash
HELL=Hello

LOCAL HELLO=World
echo $HELLO
}

function hello \
#!/bin/bash
HELL=Hello

//some text
//some text1
```

```
1 match for "LOCAL" in buffer: lab07.sh
4: LOCAL HELLO=World
```

В ходе выполнения работы я получил практические навыки по работе с редактором emacs.