

Лабораторная Работа №9

Простейший шаблон

Сильвен Макс Грегор Филс , НКАбд-03-22

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

Задание

1. Ознакомиться с теоретическим материалом.
2. Ознакомиться с редактором emacs.
3. Выполнить упражнения.
4. Ответить на контрольные вопросы.

1. Открыть emacs.
2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).
3. Наберите текст, который дан.
4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s). ## Основные команды emacs
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-/). ## Основные команды emacs
6. Научитесь использовать команды по перемещению курсора. 6.1. Переместите курсор в начало строки (C-a). 6.2. Переместите курсор в конец строки (C-e). 6.3. Переместите курсор в начало буфера (M-<). 6.4. Переместите курсор в конец буфера (M->). ##

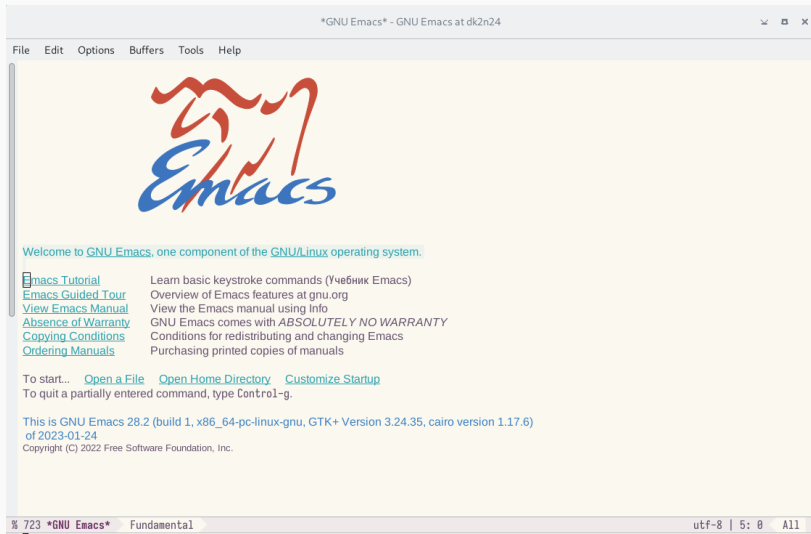
Emacs — один из наиболее мощных и широко распространённых редакторов, используемых в мире UNIX. По популярности он соперничает с редактором vi и его клонами. В зависимости от ситуации, Emacs может быть:

- текстовым редактором;
- программой для чтения почты и новостей Usenet;
- интегрированной средой разработки (IDE);
- операционной системой;

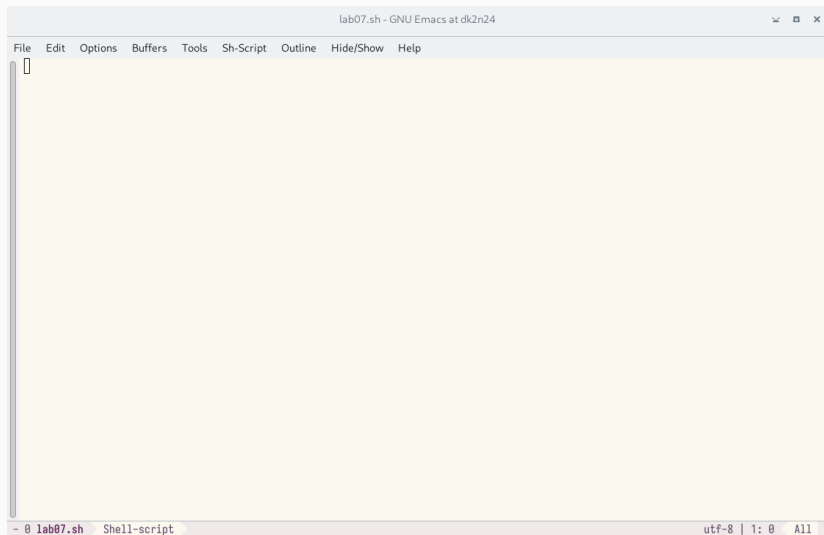
Всё это разнообразие достигается благодаря архитектуре Emacs, которая позволяет расширять возможности редактора при помощи языка Emacs Lisp. На языке C написаны лишь самые базовые и низкоуровневые части Emacs, включая полнофункциональный интерпретатор языка Lisp. Таким образом, Emacs имеет встроенный язык программирования, который может использоваться для настройки, расширения и изменения поведения редактора. В действительности, большая часть того редактора, с которым пользователи Emacs работают в наши дни, написана на языке Lisp. Первая версия редактора Emacs была написана в 70-х годах 20-го столетия Richard Stallman (Ричардом Столманом) как набор макросов для редактора TECO. В дальнейшем, уже будучи основателем Фонда Свободного программного обеспечения Free Software Foundation и проекта GNU, Stallman разработал GNU Emacs в

Выполнение лабораторной работы

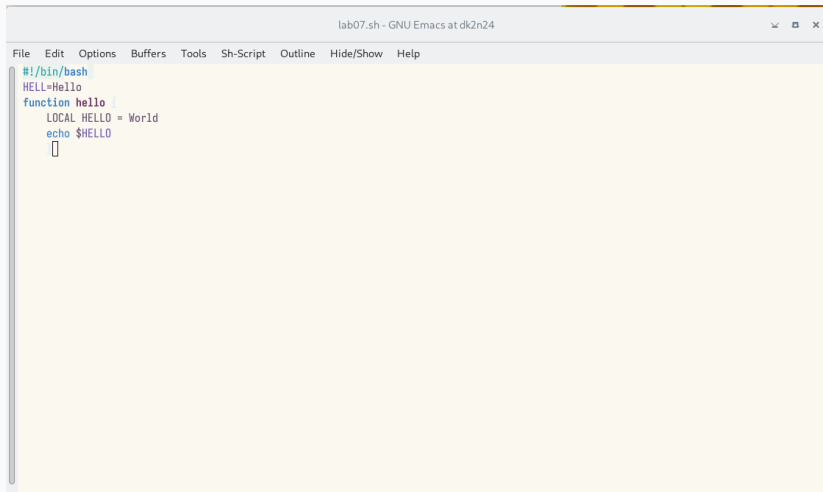
1. Открыть emacs.(рис. (fig:001?))



2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).(рис. (fig:002?))



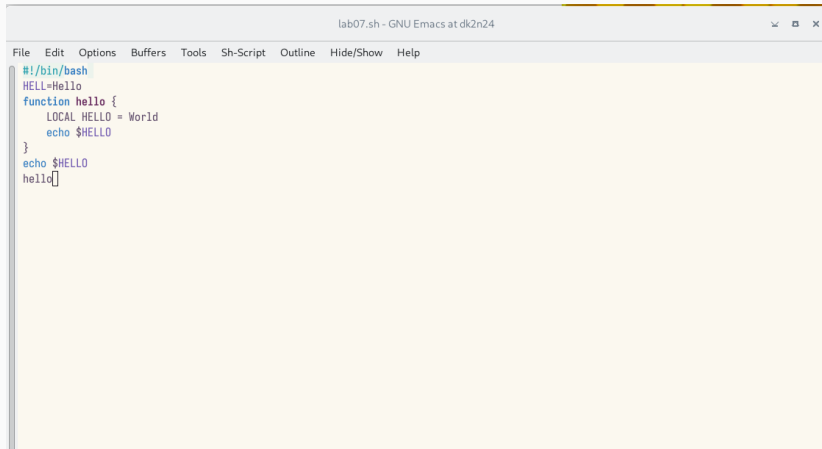
3. Наберите текст, который дан.
4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).(рис. (fig:003?))



The image shows a screenshot of the GNU Emacs editor window. The title bar at the top reads "lab07.sh - GNU Emacs at dk2n24". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", "Outline", "Hide/Show", and "Help". The main editing area contains the following shell script code:

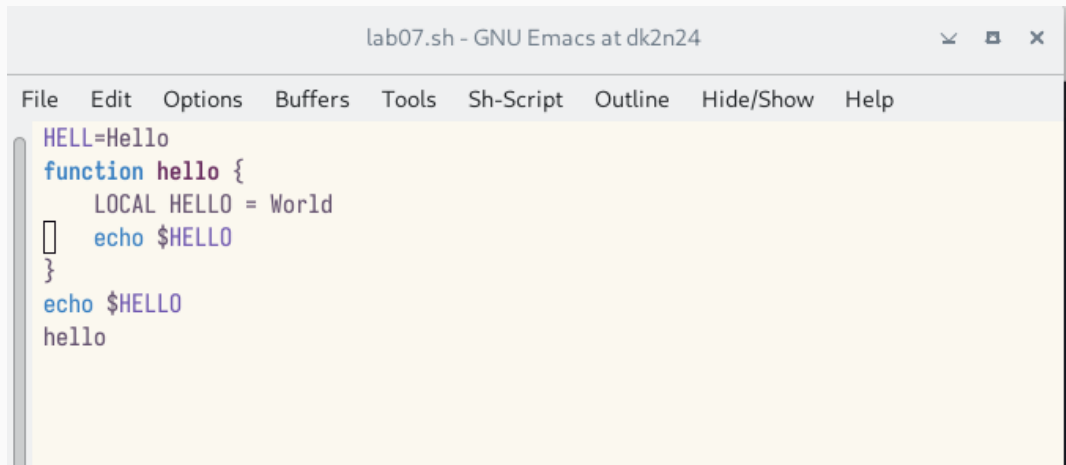
```
#!/bin/bash
HELL=Hello
function hello {
  LOCAL HELLO = World
  echo $HELLO
}
```

5. Прodelать с текстом стандартные процедуры редактирования, каждое действие должно осуществляться комбинацией клавиш. 5.1. Вырезать одной командой целую строку (C-k).(рис. (fig:004?))



```
lab07.sh - GNU Emacs at dk2n24
File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help
#!/bin/bash
HELL=Hello
function hello {
  LOCAL HELLO = World
  echo $HELLO
}
echo $HELLO
hello[]
```

5.2. Вставить эту строку в конец файла (C-y).(рис. (fig:005?))



```
lab07.sh - GNU Emacs at dk2n24

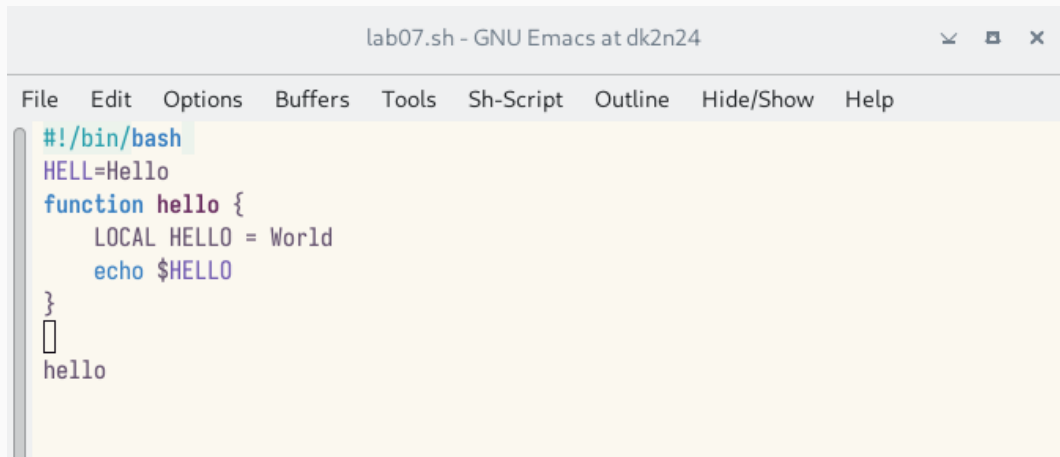
File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help

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

width=100%}

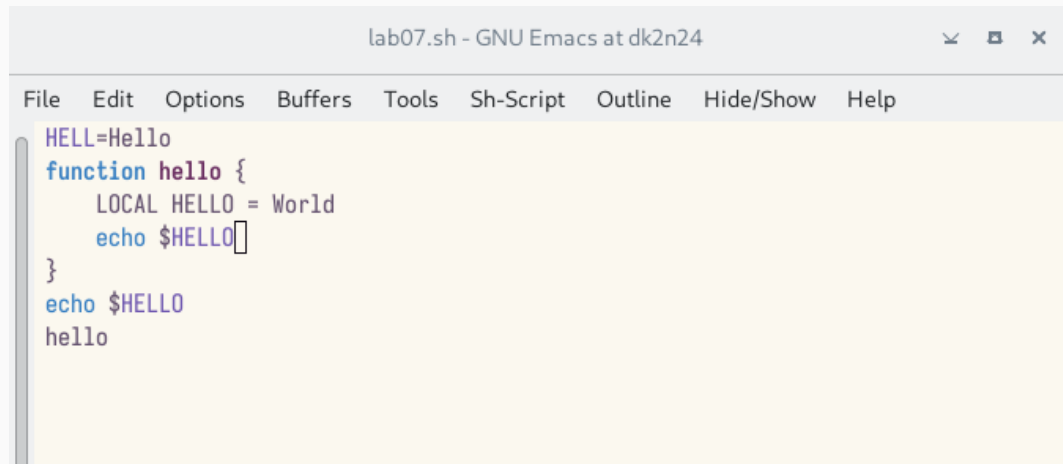
{{(fig:00

5.3. Выделить область текста (C-space). 5.4. Скопировать область в буфер обмена (M-w). 5.5. Вставить область в конец файла.(рис. (fig:006?))



```
lab07.sh - GNU Emacs at dk2n24
File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO = World
    echo $HELLO
}
hello
```

5.6. Вновь выделить эту область и на этот раз вырезать её (C-w).(рис. (fig:007?))

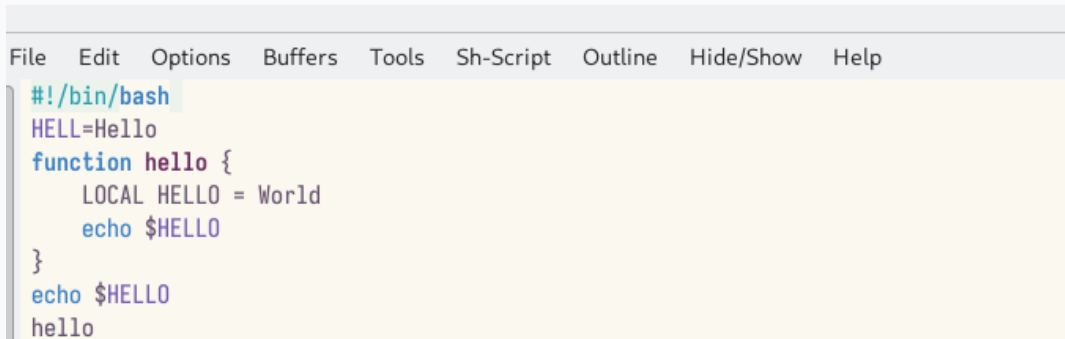


```
lab07.sh - GNU Emacs at dk2n24
File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help
HELL=Hello
function hello {
    LOCAL HELLO = World
    echo $HELLO
}
echo $HELLO
hello
```

width=100%}

{{(fig:00

5.7. Отмените последнее действие (C-/)(рис. (fig:008?))



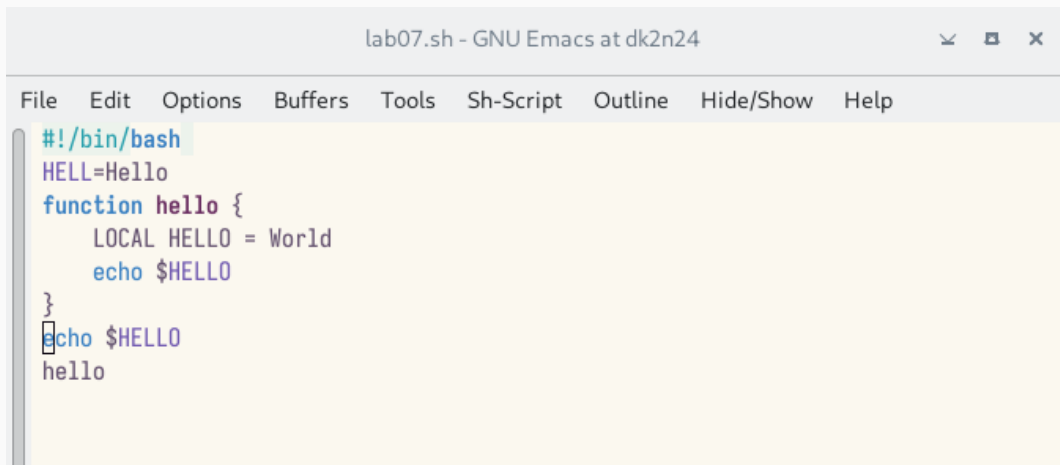
```
File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help
#!/bin/bash
HELL=Hello
function hello {
    LOCAL HELLO = World
    echo $HELLO
}
echo $HELLO
hello
```

(fig:008

width=100%}

Выполнение лабораторной работы

6. Научитесь использовать команды по перемещению курсора. 6.1. Переместите курсор в начало строки (C-a).(рис. (fig:009?))

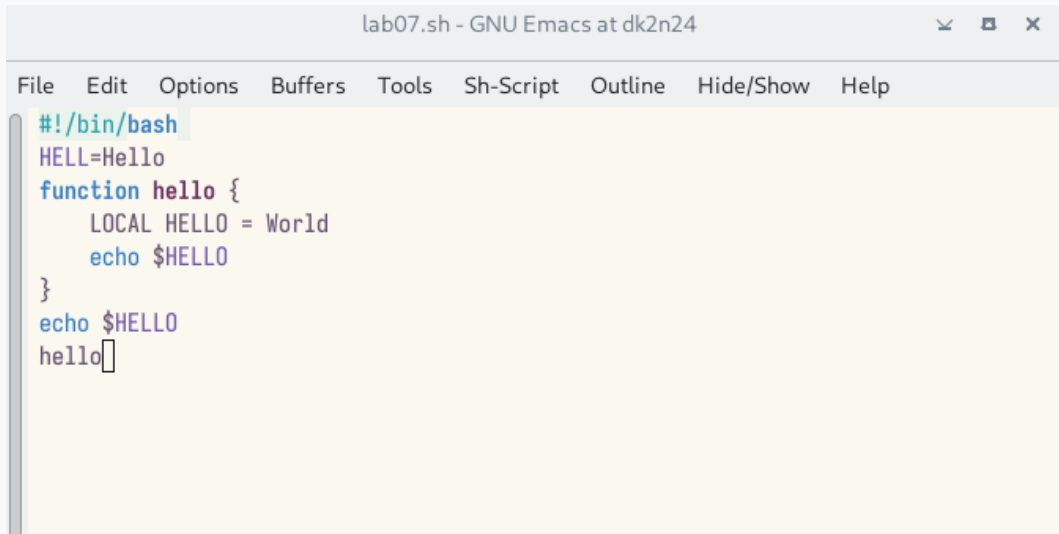


The image shows a screenshot of a GNU Emacs editor window titled "lab07.sh - GNU Emacs at dk2n24". The window has a menu bar with "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", "Outline", "Hide/Show", and "Help". The main editing area contains a shell script with the following content:

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

The cursor is positioned at the beginning of the line "echo \$HELLO".

6.2. Переместите курсор в конец строки (C-e). (рис. (fig:010?))



The image shows a screenshot of a GNU Emacs editor window titled "lab07.sh - GNU Emacs at dk2n24". The window has a menu bar with "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", "Outline", "Hide/Show", and "Help". The main text area contains a shell script with the following content:

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

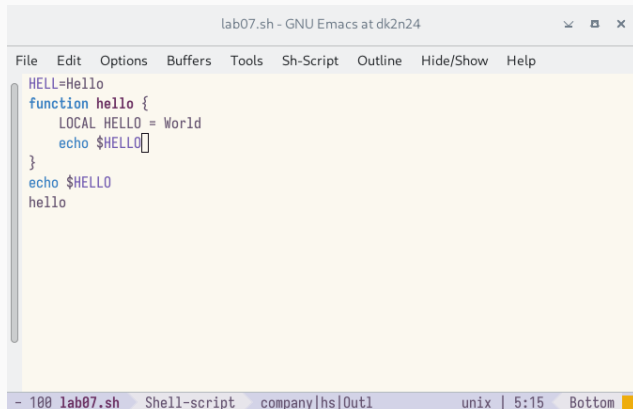
The cursor is positioned at the end of the "hello" line, indicated by a small white rectangle.

Выполнение лабораторной работы

6.3. Переместите курсор в начало буфера (M-<). 6.4. Переместите курсор в конец буфера (M->).

7. Управление буферами.

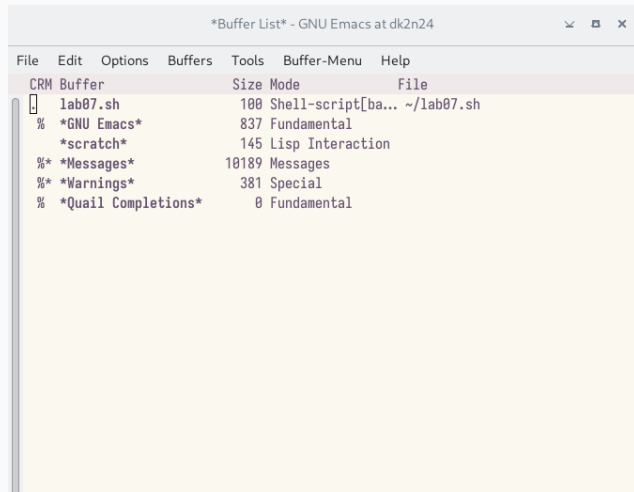
7.1. Вывести список активных буферов на экран (C-x C-b).(рис. (fig:011?))



The screenshot shows the GNU Emacs editor window titled "lab07.sh - GNU Emacs at dk2n24". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Sh-Script", "Outline", "Hide/Show", and "Help". The "Buffers" menu is open, displaying a list of active buffers: "lab07.sh", "Shell-script", "company|hs|Outl", "unix", and "Bottom". The main text area contains the following shell script code:

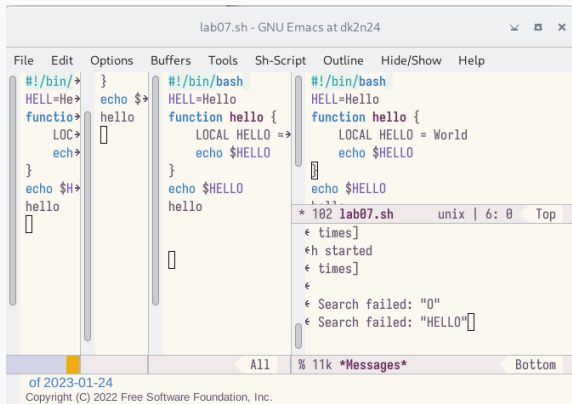
```
HELL=Hello
function hello {
  LOCAL HELLO = World
  echo $HELLO
}
echo $HELLO
hello
```

7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых буфферов и переключитесь на другой буфер.(рис. (fig:0012?))



Выполнение лабораторной работы

7.3. Закройте это окно (C-x 0). 7.4. Теперь вновь переключайтесь между буферами, но уже без вывода их списка на экран (C-x b). 8. Управление окнами. 8.1. Поделите фрейм на 4 части: разделите фрейм на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по горизонтали (C-x 2). (рис. (fig:0013?))



```
lab07.sh - GNU Emacs at dk2n24
```

File Edit Options Buffers Tools Sh-Script Outline Hide/Show Help

```
#!/bin/→  
HELL=He→  
functio→  
  LOC→  
  ech→  
}  
echo $H→  
hello  
[
```

```
}  
echo $→  
hello  
[
```

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

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

* 102 lab07.sh unix | 6: 0 Top

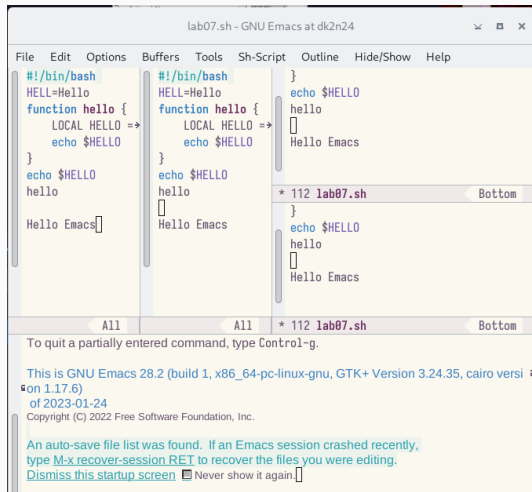
```
← times]  
← h started  
← times]  
←  
← Search failed: "0"  
← Search failed: "HELLO" [
```

All % 11k *Messages* Bottom

of 2023-01-24
Copyright (C) 2022 Free Software Foundation, Inc.

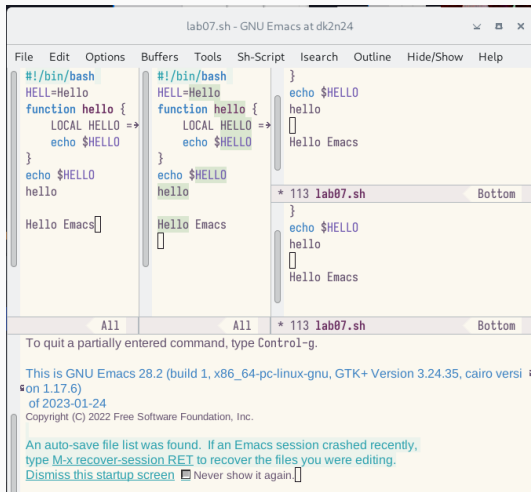
Выполнение лабораторной работы

8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста. (рис. (fig:0014?))



Выполнение лабораторной работы

9. Режим поиска 9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте.(рис. (fig:0015?))



9.2. Переключайтесь между результатами поиска, нажимая C-s.(рис. (fig:0016?))



Выполнение лабораторной работы

9.3. Выйдите из режима поиска, нажав C-g. 9.4. Перейдите в режим поиска и замены (M-%), введите текст, который следует найти и заменить, нажмите Enter , затем введите текст для замены. После того как будут подсвечены результаты поиска, нажмите ! для подтверждения замены.(рис. (fig:0017?))



The screenshot shows the GNU Emacs editor interface with a menu bar (File, Edit, Options, Buffers, Tools, Sh-Script, Isearch, Outline, Hide/Show, Help) and a toolbar. The main window displays a shell script file named 'lab07.sh' with the following content:

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

Hello Emacs
```

The search and replace operation is in progress. The search string 'HELL=Hello' is highlighted in red. The replacement string 'Hello' is highlighted in green. The search results are shown in the right-hand pane, which is titled '* 113 lab07.sh' and 'Bottom'. The search results are:

```
}
echo $HELLO
hello
[]
Hello Emacs
```

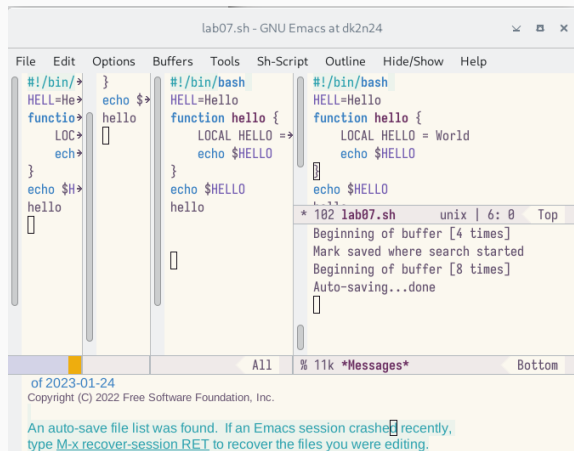
The status bar at the bottom of the window shows 'All' and '* 113 lab07.sh'.

To quit a partially entered command, type Control-g.

This is GNU Emacs 28.2 (build 1, x86_64-pc-linux-gnu, GTK+ Version 3.24.35, cairo versi on 1.17.6) of 2023-01-24

Выполнение лабораторной работы

9.5. Испробуйте другой режим поиска, нажав M-s o. Объясните, чем он отличается от обычного режима? Отличие от обычного режима в том, что тут появляется отдельное окно с текстом из файла с выделенными словами, которые нужно было найти.(рис. (fig:0018?))



Выводы

В процессе выполнения лабораторной работы я получила практические навыки работы в редакторе Emacs.

Спасибо за ваше внимание !