

Отчет по лабораторной работе №11

Операционные системы

Петрова Алевтина Александровна

Содержание

1 Цель работы

Цель данной лабораторной работы - познакомиться с операционной системой Linux. Получить практические навыки работы с редактором Emacs.

2 Задание

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

3 Теоретическое введение

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;

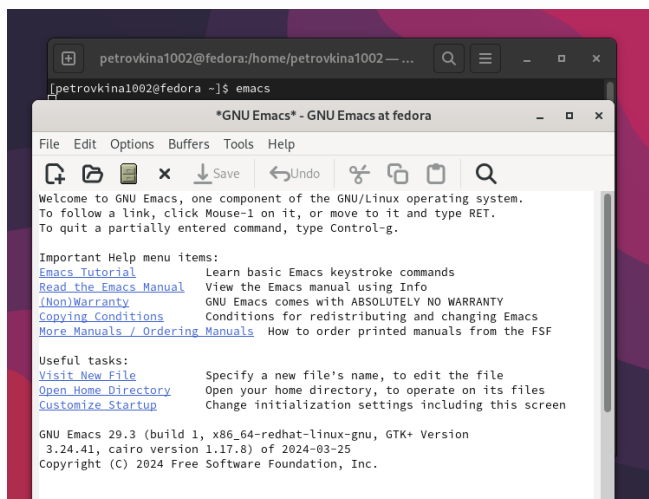
Всё это разнообразие достигается благодаря архитектуре Emacs, которая позволяет расширять возможности редактора при помощи языка Emacs Lisp. На языке C написаны лишь самые базовые и низкоуровневые части Emacs, включая полнофункциональный интерпретатор языка Lisp. Таким образом, Emacs имеет встроенный 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 развитие оригинального Emacs и до сих пор сопровождает эту программу. Emacs является одним из старейших редакторов. Он использовался тысячами программистов на протяжении последних 20 с лишним лет, для него создано много дополнительных пакетов расширений. Эти дополнения позволяют делать с помощью Emacs такие вещи, которые Stallman, вероятно, даже не считал возможными в начале своей работы над редактором.

4 Выполнение лабораторной работы

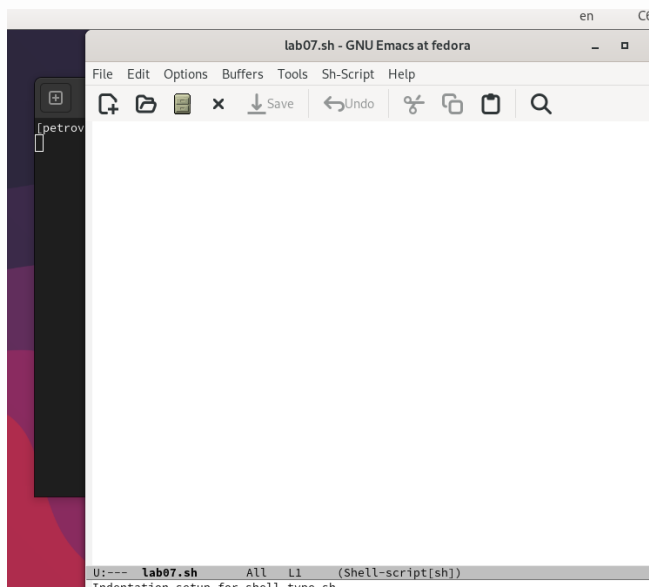
Сначала я установила Emacs.

Открываю Emacs через терминал (рис.1).



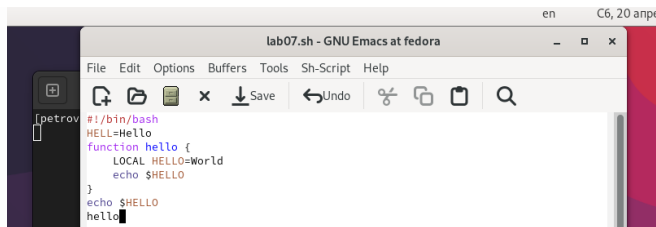
Открытие программы

Создаю файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (рис.2).



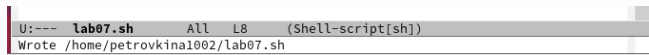
Создание файла

Прописываю в файле текст программы (рис.3).



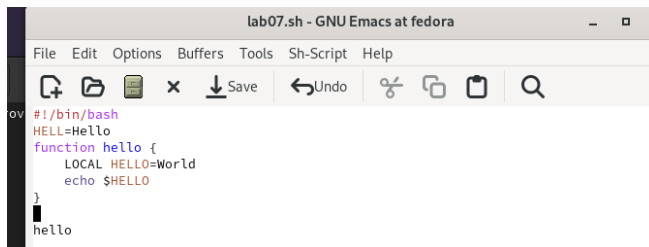
Редактирование файла

Сохраняю файл с помощью комбинации C-x C-s (рис.4).



Сохранение изменений в файле

Вырезаю одной командой целую строку (C-k) (рис.5).



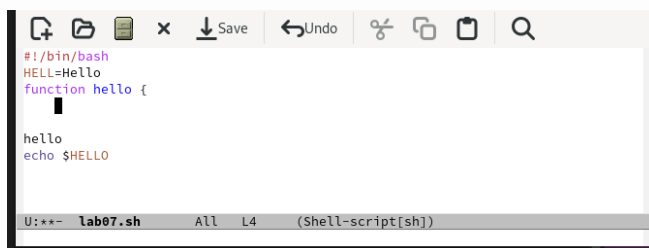
Вырезание строки

Вставляю эту строку в конец файла (C-y) (рис.6).



Вставка строки в конце файла

Выделяю область текста (C-space), копирую область в буфер обмена (M-w), вырезаю эту область с помощью C-w (рис.7).



Вырезанная область

Вставляю область в конец файла (C-y)(рис.8).



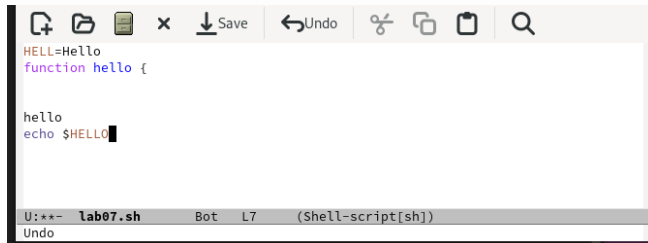
```
HELL=Hello
function hello {

hello
echo $HELLO
LOCAL HELLO=World
echo $HELLO
}

U:*** lab07.sh Bot L10 (Shell-script[sh])
```

Вставка в конец файла

Отменяю последнее действие C-/ (рис.9).



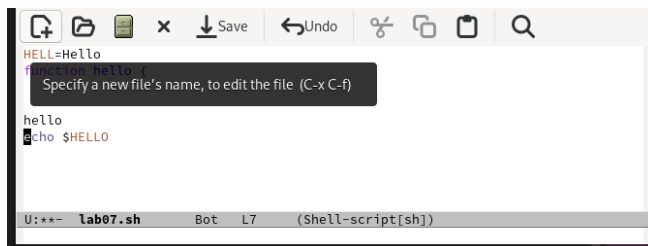
```
HELL=Hello
function hello {

hello
echo $HELLO

U:*** lab07.sh Bot L7 (Shell-script[sh])
Undo
```

Отмена последнего действия

Перевожу курсор в начало строки C-а (рис.10).



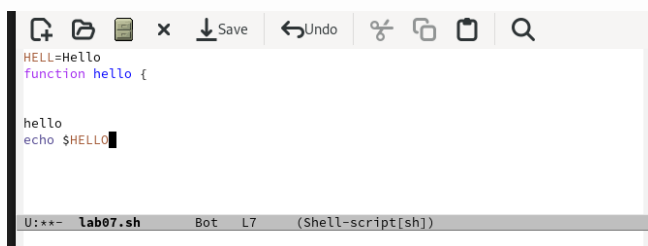
```
HELL=Hello
function hello {
hello
echo $HELLO

U:*** lab07.sh Bot L7 (Shell-script[sh])
```

Specify a new file's name, to edit the file (C-x C-f)

Курсор в начале строки

Перемещаю курсор в конец строки C-е (рис. 11).



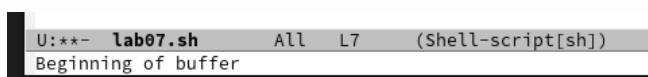
```
HELL=Hello
function hello {

hello
echo $HELLO

U:*** lab07.sh Bot L7 (Shell-script[sh])
```

Курсор в конце строки

Перемещаю курсор в начало файла M-< (рис.12).



```
U:*** lab07.sh All L7 (Shell-script[sh])
Beginning of buffer
```

Начало буфер

Перемещаю курсор в конец файла M→(рис. 13).

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

hello
echo $HELLO

U:**- lab07.sh All L7 (Shell-script[sh])
Visit tags table (default TAGS): ~/
```

Конец буфера

Открываю список активных буферов в другом окне C-x C-b (рис.14).

```
U:**- lab07.sh Bot L7 (Shell-script[sh])
CRM Buffer Size Mode File
% * lab07.sh 63 Shell-script[sh] ~/lab07.sh
% *GNU Emacs* 904 Fundamental
% *scratch* 145 Lisp Interaction
U:%%- *Buffer List* Top L1 (Buffer Menu)
Visit tags table (default TAGS): ~/
```

Список активных буферов

Переключаюсь на другой буфер C-x o (рис.15).

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

Important Help menu items:
Emacs Tutorial Learn basic Emacs keystroke commands
Read the Emacs Manual View the Emacs manual using Info
\(Non\)Warranty GNU Emacs comes with ABSOLUTELY NO WARRANTY
Copying Conditions Conditions for redistributing and changing E
More Manuals / Ordering Manuals How to order printed manuals from t
[]
Useful tasks:
U:%%- *GNU Emacs* 16% L11 (Fundamental)
visit-tags-table-buffer: File /home/petrovkina1002/TAGS does not exi
C-x <down> is undefined
End of buffer [2 times]

U:%%- *Messages* Bot L38 (Messages)
```

Другое окно буфера

Закрываю окно другого буфера C-x 0 (рис.16).

```
CRM Buffer Size Mode File
% *Async-native-compile-... 165 Fundamental
% *GNU Emacs* 904 Fundamental
% *Messages* 1398 Messages
% *scratch* 145 Lisp Interaction
. * lab07.sh 63 Shell-script[sh] ~/lab07.sh

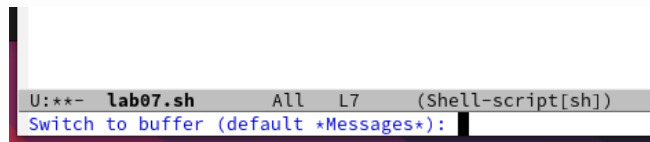
U:%%- *Buffer List* All L1 (Buffer Menu)
#!/bin/bash
HELL=Hello
function hello {

hello
echo $HELLO

U:**- lab07.sh All L1 (Shell-script[sh])
```

Заккрытие буфера

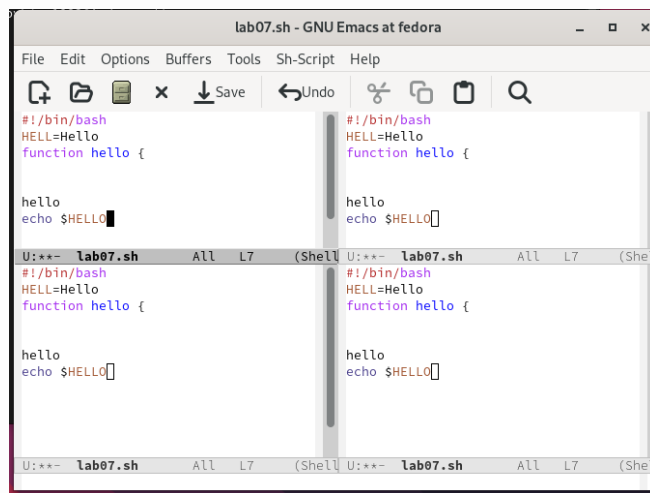
Открываю другой буфер без вывода их списка на экран с помощью C-x b(рис.17).

A screenshot of the Emacs command line showing the command 'Switch to buffer (default *Messages*):' with a cursor at the end of the line.

```
U:*** lab07.sh All L7 (Shell-script[sh])
Switch to buffer (default *Messages*):
```

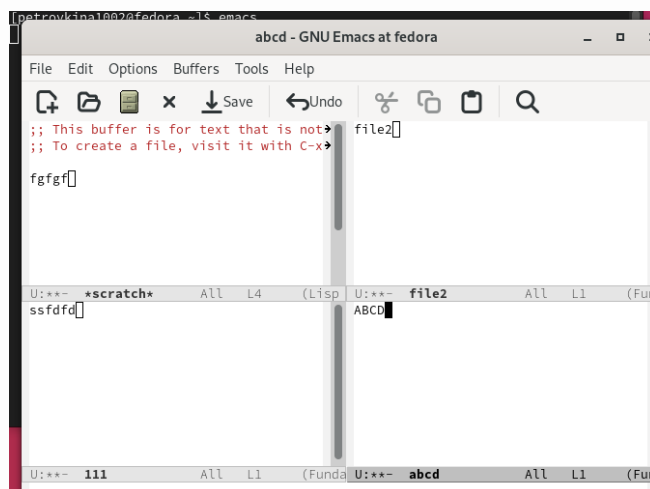
Открытие другого буфера

Делю фрейм на 4 части: сначала на два окна по вертикали (C-x 3), а затем каждое из этих окон на две части по горизонтали (C-x 2) (рис.18).



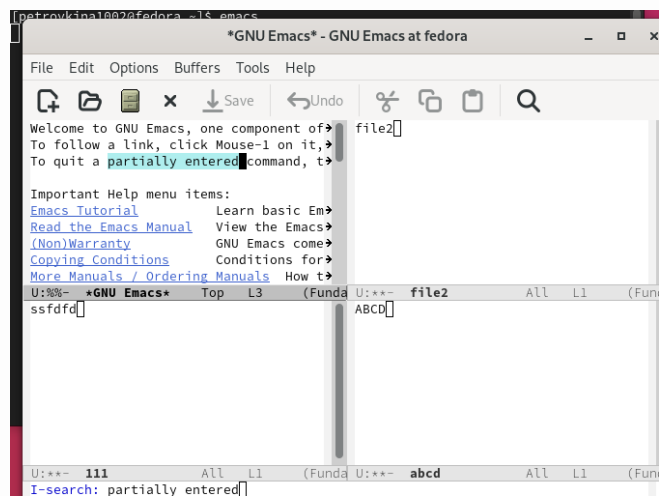
Четыре окна

В каждом из четырех созданных окон с помощью C-x b открыла разные буферы и (рис.19).



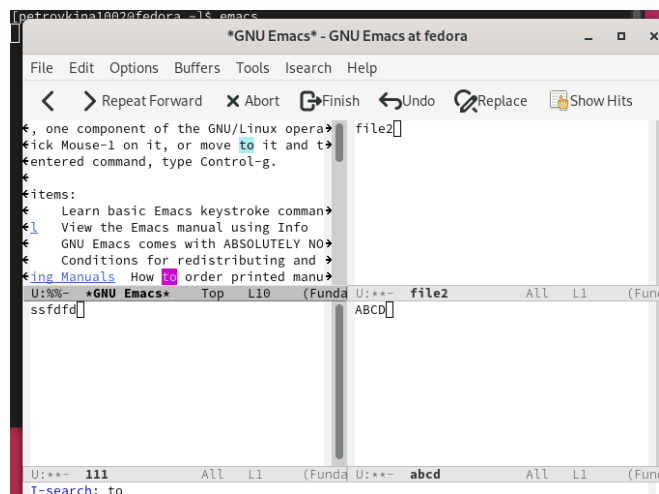
Открытие буферов в четырех окнах

Перехожу в режим поиска с помощью C-s, ищу слова в тексте, они подсвечиваются (рис.20).



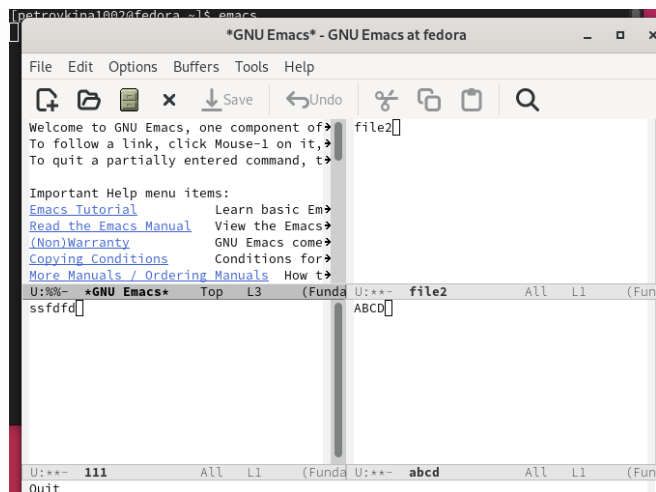
Режим поиска

С помощью той же комбинации C-s я могу перемещаться по результатам поиска (рис.21).



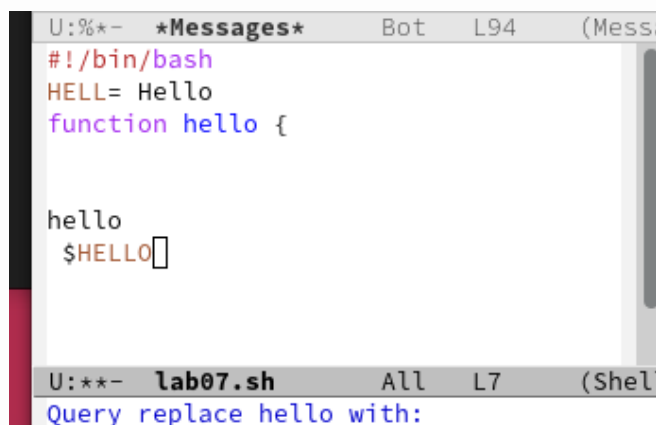
Перемещение по найденным выражениям

С помощью C-g выхожу из режима поиска, снимается выделение (рис.22).



Выход из режима поиска

Перехожу в режим поиска и замены с помощью M-%, ввожу какое слово хочу заменить, затем ввожу на какое хочу заменить (рис.23).



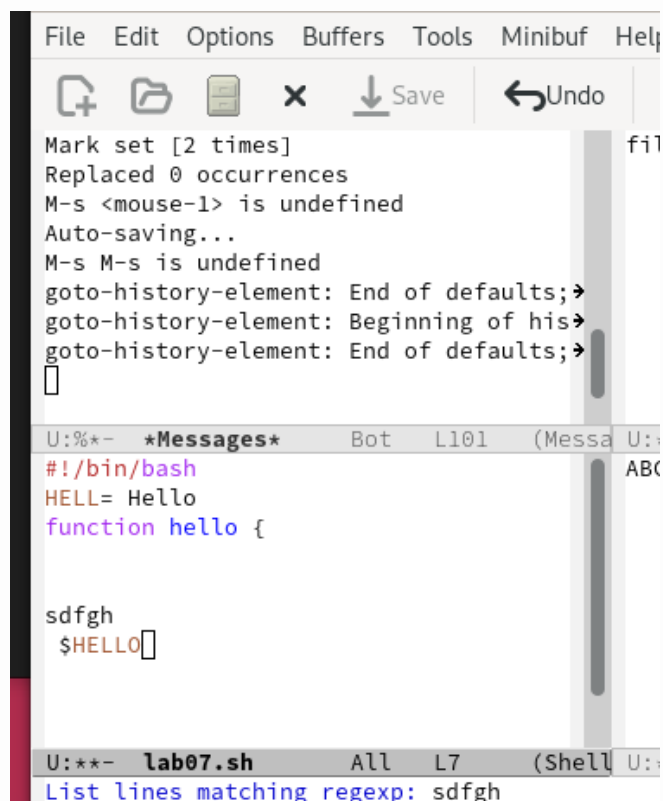
Замена слова

Видим, что слова были заменены успешно (рис.24).



Слово заменено

С помощью M+s перехожу в другой режим поиска. Он отличается от предыдущего тем, что выводит результат в отдельном окне от окна буфера (рис.25).



Режим поиска

5 ВЫВОДЫ

В ходе данной лабораторной работы я получила практические навыки работы с редактором Emacs.

6 Ответы на контрольные вопросы

- ## 1. Кратко охарактеризуйте редактор emacs.

Emacs — один из наиболее мощных и широко распространённых редакторов, используемых в мире UNIX. Написан на языке высокого уровня Lisp.

1. Какие особенности данного редактора могут сделать его сложным для освоения новичком?

Большое разнообразие сложных комбинаций клавиш, которые необходимы для редактирования файла и в принципе для работы с Emacs.

1. Своими словами опишите, что такое буфер и окно в терминологии emacs'а.

Буфер - это объект в виде текста. Окно - это прямоугольная область, в которой отображен буфер.

1. Можно ли открыть больше 10 буферов в одном окне?

Да, можно.

1. Какие буферы создаются по умолчанию при запуске emacs?

Emacs использует буферы с именами, начинающимися с пробела, для внутренних целей. Отчасти он обращается с буферами с такими именами особым образом — например, по умолчанию в них не записывается информация для отмены изменений.

1. Какие клавиши вы нажмёте, чтобы ввести следующую комбинацию C-c | и C-c C-|?

Ctrl + c, а потом | и Ctrl + c Ctrl + |

1. Как поделить текущее окно на две части?

С помощью команды Ctrl + x 3 (по вертикали) и Ctrl + x 2 (по горизонтали).

1. В каком файле хранятся настройки редактора emacs?

Настройки emacs хранятся в файле `.emacs`, который хранится в домашней директории пользователя. Кроме этого файла есть ещё папка `.emacs`.

1. Какую функцию выполняет клавиша и можно ли её переназначить?

Выполняет функцию стереть, думаю можно переназначить.

1. Какой редактор вам показался удобнее в работе vi или emacs? Поясните почему.

Для меня удобнее был редактор Emacs, так как у него есть командная оболочка. А vi открывается в терминале, и выглядит своеобразно.