---

# Front matter

lang: ru-RU

title: "Отчет по лабораторной работе 10"

subtitle: "Операционные системы"

author: "Дорофеева Алёна Тимофеевна"

group: "НПИбд-01-20"

# Formatting

fontsize: 12pt

linestretch: 1.5

papersize: a4paper

---

\*\*Теоретические вводные данные\*\* [[1\]](http://www.linuxcookbook.ru/books/slackbook/emacs.html) :

В то время как vi является без сомнения самым распространённым редактором в Unix-подобных системах, \*\*Emacs\*\* считается вторым хорошим редактором. Вместо различных “\*режимов\*”, применяемых в vi, для ввода команд в нём используются комбинации клавиш с `Control` и `Alt`.

Также, в отличие от vi, \*\*Emacs\*\* - это программа с практически неограниченными возможностями. \*\*Emacs\*\* (по большей части) написан на \*Lisp\* - очень мощном языке программирования, характерной особенностью которого является то, что любая написанная на этом языке программа автоматически сама для себя является компилятором Lisp. Это означает, что пользователь может самостоятельно расширять возможности \*\*Emacs\*\*, а по сути писать новые программы “на Emacs'е”.

Каждый \*буфер\* в \*\*Emacs\*\* имеет связанный с ним режим. Этот режим очень отличается от идеи режимов в vi: режим говорит вам о типе буфера, в котором вы находитесь. Например, для обычных текстовых файлов существует \*text-mode\* (текстовый режим), однако также существуют такие режимы как: \*c-mode\* (режим C) для редактирования программ на C, \*sh-mode\* (режим командного процессора) для редактирования шелл-скриптов, \*latex-mode\* (режим latex) для редактирования файлов LaTeX, \*mail-mode\* (почтовый режим) для редактирования электронных писем и новостей и другие режимы. Режимы предоставляют специальные возможности и функции, полезные для различных типов редактируемых файлов. Для режимов даже можно переопределять клавиши и клавишные команды. Например, в режиме текста клавиша `Tab` выполняет перескакивание на окончание следующего табулятора, а в большинстве режимов для языков программирования клавиша Tab делает отступ текущей строки на соответствующую глубину текущего блока, в котором находится эта строка.

Упомянутые выше режимы относятся к главным режимам. У каждого буфера есть только один главный режим. Дополнительно буфер может иметь один или несколько второстепенных режимов. Второстепенный режим предоставляет дополнительные возможности, которые могут быть полезными в определённых задачах редактирования. Например, если вы нажмёте клавишу `INSERT`, вы включите режим \*замены\*, т.е. именно то, что вы и ожидали от этой клавиши. Есть ещё режим \*автозаполнения\* (auto-fill), который удобен в комбинации с режимами текста или latex: в нём каждая набранная вами строка будет автоматически разбиваться с переносом на следующую строку при достижении определённого количества символов.

\*\*Условные обозначения и символы\*\*:

С - клавиша `ctrl`

M - клавиша `meta (alt)`

S - клавиша `shift`

# Цель работы

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

# Задачи

- Ознакомиться с теоретическим материалом

- Ознакомиться с редактором emacs

- Выполнить упражнения

- Ответить на контрольные вопросы

# Выполнение лабораторной работы

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию

1. Открыла emacs (рис. 1) и создала файл lab07.sh с помощью комбинации клавиш ``Ctrl-x`` ``Ctrl-f`` , набираю текст и сохраняю файл с помощью комбинации ``Ctrl-x`` ``Ctrl-s`` (рис. 2)

<img src="скрины10\1.PNG">

\*Рис. 1: Открыла emacs\*

<img src="скрины10\2.PNG">

\*Рис. 2: Создала файл при помощи комбинации клавиш, ввела текст и сохранила\*

2. Проделываю с текстом стандартные процедуры редактирования. Вырезаю одной комбинацией целую строку ``Ctrl-k`` (рис. 3)

<img src="скрины10\3.PNG">

\*Рис. 3: Вырезаю строку\*

3. Вставляю эту строку в конце файла ``Ctrl-y``(рис. 4)

<img src="скрины10\4.PNG">

\*Рис. 4: Вставляю вырезанную строку в конец файла\*

4. Выделяю область текста ``Ctrl-Space`` (рис. 5), скопировала область в буфер обмена ``M-w`` и вставила область в конец файла (рис. 6)

<img src="скрины10\5.PNG">

\*Рис. 5: Выделенная область\*

<img src="скрины10\6.PNG">

\*Рис. 6: Вставила в конец файла скопированную область\*

5. Вновь выделяем область и вырезаем ее ``C-w`` (рис. 7)

<img src="скрины10\7.PNG">

\*Рис. 7: Выделила область и вырезала ее\*

6. Отменила последнее действие ``C-/`` (рис. 8)

<img src="скрины10\8.PNG">

\*Рис. 8: Отменила последнее действие\*

7. Научилась использовать команды по перемещению курсора: переместила курсор в начало строки клавишей `C-a` , переместила курсор в конец строки клавишей `C-e` , переместила курсор в начало буфера клавишей `M-< ` , переместила курсор в конец буфера клавишей `M-> ` .

8. Вывела список активных буфером на экран клавишей `C-x` `C-b` и закрыла это окно клавишами `C-x` `0`. (рис. 9)

<img src="скрины10\9.PNG">

\*Рис. 9: Список активных буферов\*

9. Переместилась во вновь открытое окно (`C-x`) o со списком открытых буферов и переключитесь на другой буфер. (рис. 10) Возвращаюсь в файл lab07.sh `C-x b`. (рис. 11)

<img src="скрины10\10.PNG">

\*Рис. 10: Переключение на другой буфер\*

<img src="скрины10\11.PNG">

\*Рис. 11: Возвращаемся в lab07\*

10. Поделила 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. 12)

<img src="скрины10\12.PNG">

\*Рис. 12: Деление фрейма\*

11. В одном из созданных окон открываю новый буфер и ввела текст (рис. 13 и 14)

<img src="скрины10\13.1.PNG">

\*Рис. 13: Создание нового буфера\*

<img src="скрины10\13.2.PNG">

\*Рис. 14: Ввод нового текста\*

12. Переключаюсь в режим поиска ``C-s`` , переключаемся между результатами с помощью этой же комбинацией (рис. 15)

<img src="скрины10\14.PNG">

\*Рис. 15: Переходим в режим поиска\*

13. Выходим из режима поиска, нажимая ``C-g`` (рис. 16)

<img src="скрины10\15.PNG">

\*Рис. 16: Выходим из режима поиска\*

14. Переходим в режим поиска и замены ``M-%`` , вводим текст, который который следует найти и заменить, нажмите ``Enter`` , затем вводим текст для замены (рис. 17) После того как будут подсвечены результаты поиска (рис. 18), нажимаем ``!`` для подтверждения замены. (рис. 19)

<img src="скрины10\16.PNG">

\*Рис. 17: Вводим текст, который нужно найти и заменить\*

<img src="скрины10\17.PNG">

\*Рис. 18: Видим подсвеченные результаты поиска\*

<img src="скрины10\18.PNG">

\*Рис. 19: Замена воспроизведена\*

15. Пробую новый режим поиска, нажав сочетание клавиш `M-s o` . От обычного поиска он отличается тем, что есть возможность поиска по регулярным выражениям. Результаты выводятся в отдельном окне с указанием количества вхождений.

<img src="скрины10\19.PNG">

# Вывод

Я познакомилась с операционной системой \*\*Linux\*\*. Получила практические навыки работы с редактором Emacs.

\_\_\_

### \*\*Библиографический список\*\*:

[1]: [Редактор emacs](http://www.linuxcookbook.ru/books/slackbook/emacs.html)

[2]: [Описание лабораторной работы №10](https://esystem.rudn.ru/pluginfile.php/1142374/mod\_resource/content/3/007-lab\_emacs.pdf)