

Отчет

Лабораторная работа №9

Павлова Варвара Юрьевна НПМбд-02-21

Содержание

Цель работы	5
Задание	6
Теоретическое введение	8
Выполнение лабораторной работы	9
Выводы	19
Список литературы	20

Список иллюстраций

0.1	установка emacs	9
0.2	запуск emacs	9
0.3	открытие созданного файла	10
0.4	набранный текст	11
0.5	текст с вырезанной первой строкой	11
0.6	вставка строки в конец файла	12
0.7	выделение области текста	12
0.8	вставка скопированного текста	12
0.9	вырезка текста	13
0.10	отмена последнего действия	13
0.11	курсор в начале буфера	13
0.12	список активных буферов	14
0.13	переключение между буферами	14
0.14	разделенный фрейм	15
0.15	новые буферы и текст в них	16
0.16	поиск слова text	16
0.17	переключение между результатами поиска	17
0.18	выход из режима поиска	17
0.19	режим поиска	18

Список таблиц

Цель работы

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.

Основные команды emacs

1. Открыть emacs.
2. Создать файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).
3. Наберите текст.
4. Сохранить файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).
5. Прodelать с текстом стандартные процедуры редактирования, каждое действие долж- но 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-/).
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->).
7. Управление буферами. 7.1. Вывести список активных буферов на экран (C-x C-b). 7.2. Переместитесь во вновь открытое окно (C-x) о со списком открытых

- буферов и переключитесь на другой буфер. 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). 8.2. В каждом из четырёх созданных окон откройте новый буфер (файл) и введите несколько строк текста.
9. Режим поиска 9.1. Переключитесь в режим поиска (C-s) и найдите несколько слов, присутствующих в тексте. 9.2. Переключайтесь между результатами поиска, нажимая C-s. 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. 9.5. Испробуйте другой режим поиска, нажав M-s o. Объясните, чем он отличается от обычного режима?

Теоретическое введение

Определение 1. Буфер — объект, представляющий какой-либо текст. Буфер может содержать что угодно, например, результаты компиляции программы или встроенные подсказки. Практически всё взаимодействие с пользователем, в том числе интерактивное, происходит посредством буферов.

Определение 2. Фрейм соответствует окну в обычном понимании этого слова. Каждый фрейм содержит область вывода и одно или несколько окон Emacs.

Определение 3. Окно — прямоугольная область фрейма, отображающая один из буферов. Каждое окно имеет свою строку состояния, в которой выводится следующая информация: название буфера, его основной режим, изменялся ли текст буфера и как далеко вниз по буферу расположен курсор. Каждый буфер находится только в одном из возможных основных режимов. Существующие основные режимы включают режим Fundamental(наименее специализированный), режим Text, режим Lisp, режим C, режим Texinfo и другие. Под второстепенными режимами понимается список режимов, которые включены в данный момент в буфере выбранного окна.

Определение 4. Область вывода — одна или несколько строк внизу фрейма, в которой Emacs выводит различные сообщения, а также запрашивает подтверждения и дополнительную информацию от пользователя.

Определение 5. Минибуфер используется для ввода дополнительной информации и всегда отображается в области вывода.

Определение 6. Точка вставки — место вставки (удаления) данных в буфере.

Выполнение лабораторной работы

1. Я ознакомилась с теоретическим материалом.
2. Я ознакомилась с редактором emacs.
3. Я выполнила упражнения.

Основные команды emacs

1. Устанавливаю emacs (рис. [-@fig:001]) и открываю его.(рис. [-@fig:002])

```
[vypavlova@fedora ~]$ sudo dnf install emacs
[sudo] password for vypavlova:
Copr repo for PyCharm owned by phracek      6.4 kB/s | 3.6 kB      00:00
Copr repo for PyCharm owned by phracek      25 kB/s | 43 kB      00:01
Fedora 35 - x86_64                          27 kB/s | 20 kB      00:00
Fedora Modular 35 - x86_64                 56 kB/s | 20 kB      00:00
Fedora 35 - x86_64 - Updates                39 kB/s | 14 kB      00:00
Fedora 35 - x86_64 - Updates                1.3 MB/s | 5.1 MB     00:03
```

Рис. 0.1: установка emacs

```
[vypavlova@fedora ~]$ emacs &
[1] 6900
```

Рис. 0.2: запуск emacs

2. Создаю файл lab07.sh с помощью комбинации Ctrl-x Ctrl-f (C-x C-f).(рис. [-@fig:003])

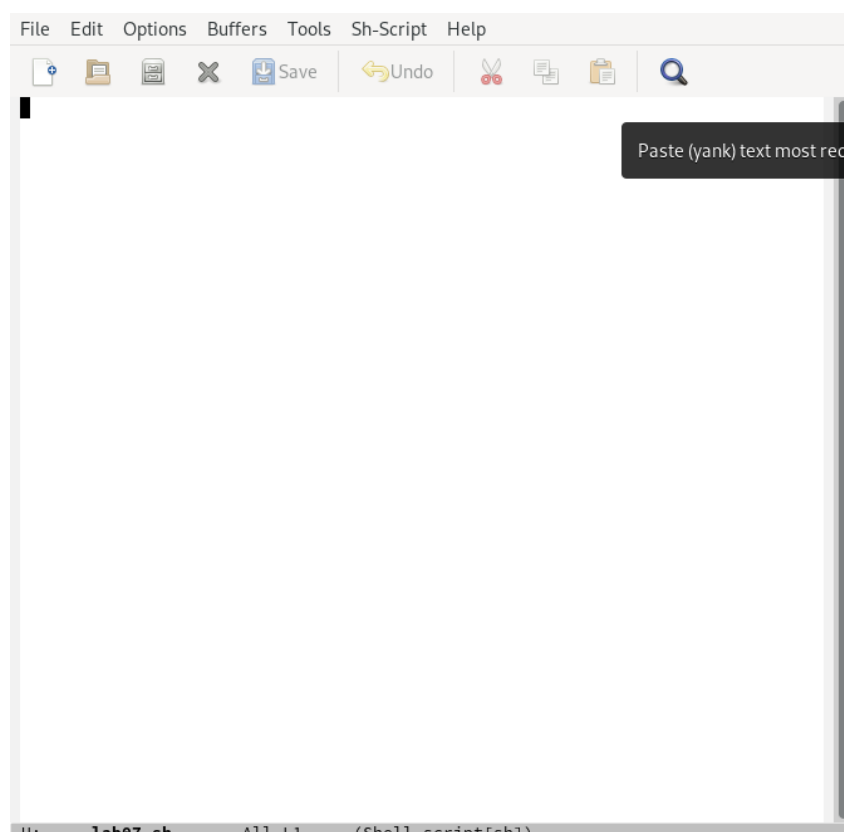


Рис. 0.3: открытие созданного файла

3. Набираю текст.(рис. [-@fig:004])

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

Рис. 0.4: набранный текст

4. Сохраню файл с помощью комбинации Ctrl-x Ctrl-s (C-x C-s).
5. Прodelьваю с текстом стандартные процедуры редактирования: 5.1. Вырезаю одной командой целую строку (C-k).(рис. [-@fig:005])

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

Рис. 0.5: текст с вырезанной первой строкой

- 5.2. Вставляю эту строку в конец файла (C-y).(рис. [-@fig:006])

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

Рис. 0.6: вставка строки в конец файла

5.3. Выделяю область текста (C-space).(рис. [-@fig:007])

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

Рис. 0.7: выделение области текста

5.4. Копирую выделенную область в буфер обмена (M-w). 5.5. Вставляю область в конец файла.(рис. [-@fig:008])

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

Рис. 0.8: вставка скопированного текста

5.6. Вновь выделяю эту область и на этот раз вырезаю её (C-w).(рис. [-@fig:009])

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

Рис. 0.9: вырезка текста

5.7. Отменяю последнее действие (C-/).(рис. [-@fig:010])

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

Рис. 0.10: отмена поседнего действия

6. Учусь использовать команды по перемещению курсора. 6.1. Перемещаю курсор в начало строки (C-a). 6.2. Перемещаю курсор в конец строки (C-e). 6.3. Перемещаю курсор в начало буфера (M-<).(рис. [-@fig:011])

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

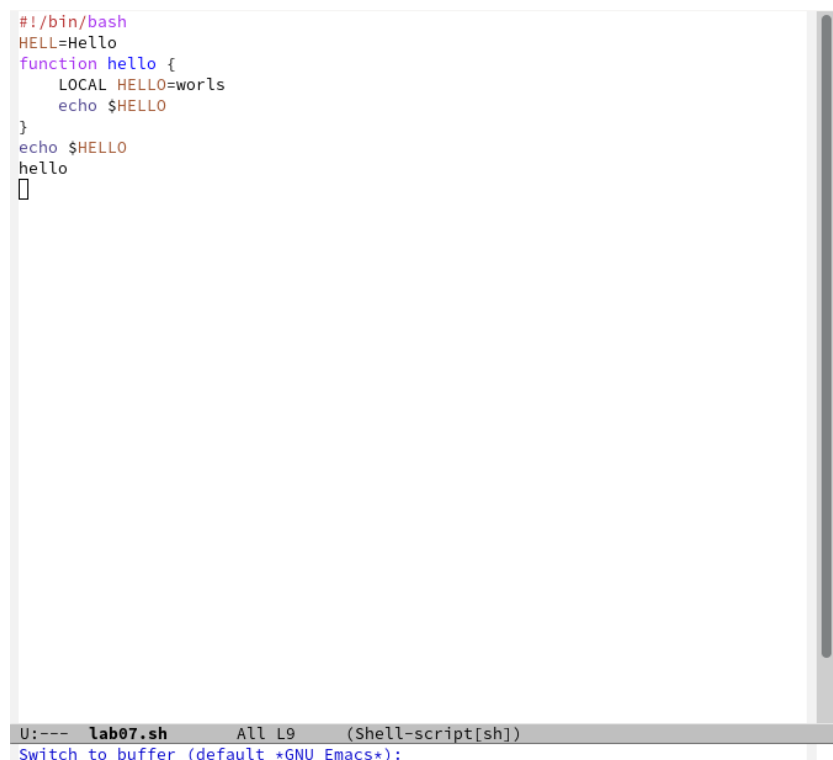
Рис. 0.11: курсор в начале буфера

6.4. Перемещаю курсор в конец буфера (M->). 7. Управление буферами. 7.1. Вывожу список активных буферов на экран (C-x C-b).(рис. [-@fig:012])

```
U:*** lab07.sh All L10 (Shell-script[sh])
CRM Buffer Size Mode File
[*] * lab07.sh 104 Shell-script[sh] ~/lab07.sh
% *GNU Emacs* 852 Fundamental
*scratch* 145 Lisp Interaction
%* *Messages* 1899 Messages
```

Рис. 0.12: список активных буферов

7.2. Перемещаюсь во вновь открытое окно (C-x o) со списком открытых буферов и переключаюсь на другой буфер. 7.3. Закрываю это окно (C-x 0). 7.4. Теперь вновь переключаюсь между буферами, но уже без вывода их списка на экран (C-x b).(рис. [-@fig:013])



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

U:---	lab07.sh	All	L9	(Shell-script[sh])
Switch to buffer (default *GNU Emacs*):				

Рис. 0.13: переключение между буферами

8. Управление окнами. 8.1. 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. [-@fig:014])

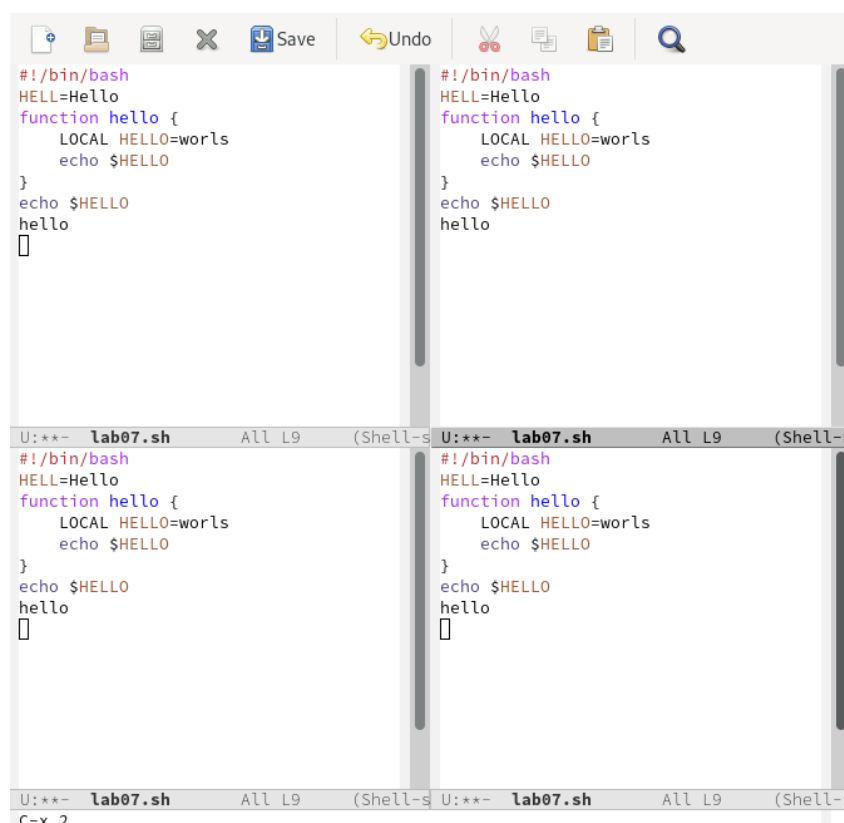


Рис. 0.14: разделенный фрейм

8.2. В каждом из четырех созданных окон открываю новый буфер (файл) и ввожу несколько строк текста.(рис. [-@fig:015])

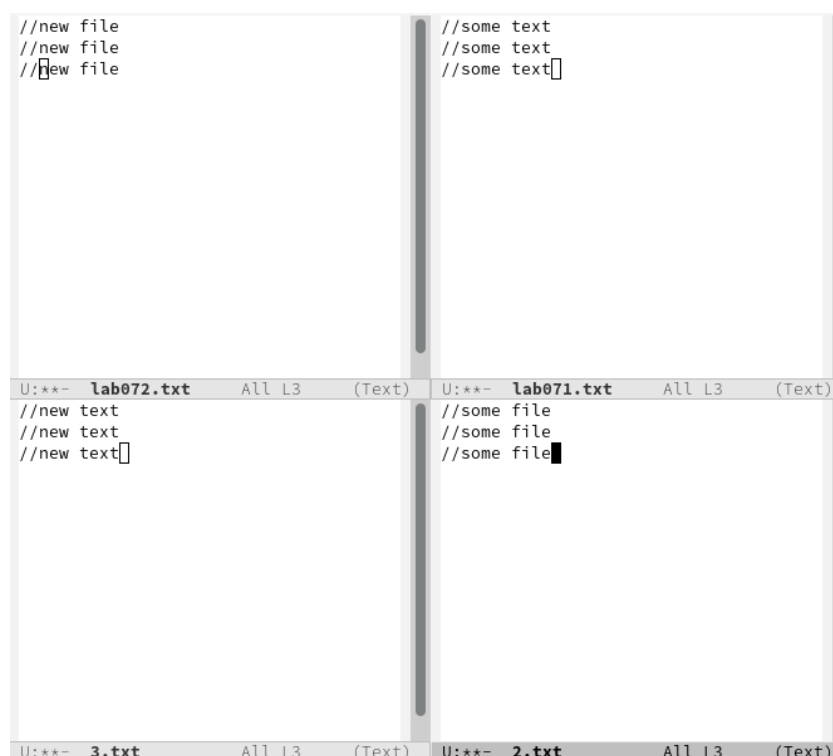


Рис. 0.15: новые буферы и текст в них

9. Режим поиска 9.1. Переключаюсь в режим поиска (C-s) и нахожу несколько слов, присутствующих в тексте.(рис. [-@fig:016])



Рис. 0.16: поиск слова text

9.2. Переключаюсь между результатами поиска, нажимая C-s.(рис. [-@fig:017])



Рис. 0.17: переключение между результатами поиска

9.3. Выхожу из режима поиска, нажав C-g.(рис. [-@fig:018])



Рис. 0.18: выход из режима поиска

9.4. К сожалению выполнение этого пункта мне не удалось, так как не распознается данная комбинация клавиш. 9.5. Пробую другой режим поиска, нажав M-s o.(рис.

[-@fig:019])

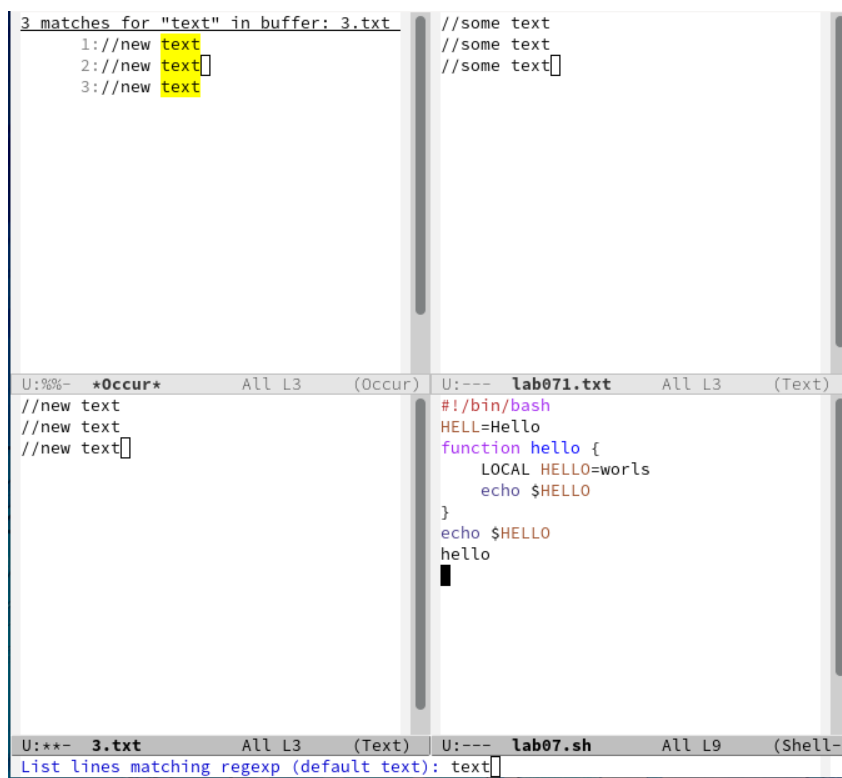


Рис. 0.19: режим поиска

Выводы

Выполняя данную лабораторную работу я приобрела навыки работы с редактором emacs и операционной системой Linux.

Список литературы