

# **Отчет по лабораторной работе №11**

**Дисциплина: Операционные системы**

Иванов Сергей Владимирович

# Содержание

|          |                                       |           |
|----------|---------------------------------------|-----------|
| <b>1</b> | <b>Цель работы</b>                    | <b>4</b>  |
| <b>2</b> | <b>Задание</b>                        | <b>5</b>  |
| <b>3</b> | <b>Выполнение лабораторной работы</b> | <b>6</b>  |
| <b>4</b> | <b>Контрольные вопросы</b>            | <b>17</b> |
| <b>5</b> | <b>Вывод</b>                          | <b>19</b> |

## Список иллюстраций

|      |   |    |
|------|---|----|
| 3.1  | Открытие программы . . . . .                  | 6  |
| 3.2  | Создание файла . . . . .                      | 6  |
| 3.3  | Редактирование файла . . . . .                | 7  |
| 3.4  | Сохранение изменений в файле . . . . .        | 7  |
| 3.5  | Вырезание строки . . . . .                    | 7  |
| 3.6  | Вставка строки в конце файла . . . . .        | 8  |
| 3.7  | Вырезанная область . . . . .                  | 8  |
| 3.8  | Вставка в конец файла . . . . .               | 9  |
| 3.9  | Отмена последнего действия . . . . .          | 9  |
| 3.10 | Курсор в начале строки . . . . .              | 10 |
| 3.11 | Курсор в конце строки . . . . .               | 10 |
| 3.12 | Начало буфера . . . . .                       | 10 |
| 3.13 | Конец буфера . . . . .                        | 11 |
| 3.14 | Список активных буферов . . . . .             | 11 |
| 3.15 | Другое окно буфера . . . . .                  | 12 |
| 3.16 | Закрытие буфера . . . . .                     | 12 |
| 3.17 | Открытие другого буфера . . . . .             | 13 |
| 3.18 | Четыре окна . . . . .                         | 13 |
| 3.19 | Открытие буферов в четырех окнах . . . . .    | 14 |
| 3.20 | Режим поиска . . . . .                        | 14 |
| 3.21 | Перемещение по найденным выражениям . . . . . | 15 |
| 3.22 | Выход из режима поиска . . . . .              | 15 |
| 3.23 | Замена слова . . . . .                        | 15 |
| 3.24 | Слово заменено . . . . .                      | 16 |
| 3.25 | Режим поиска . . . . .                        | 16 |

# 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 Выполнение лабораторной работы

Открываю Emacs через терминал (рис. 1).

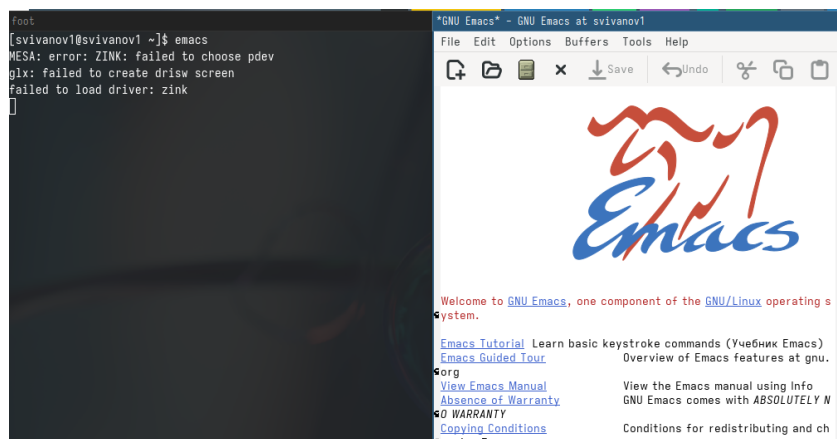


Рис. 3.1: Открытие программы

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

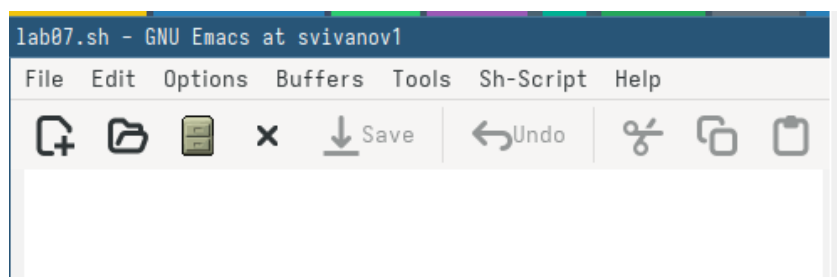


Рис. 3.2: Создание файла

Прописываю в файле текст программы (рис. 3).

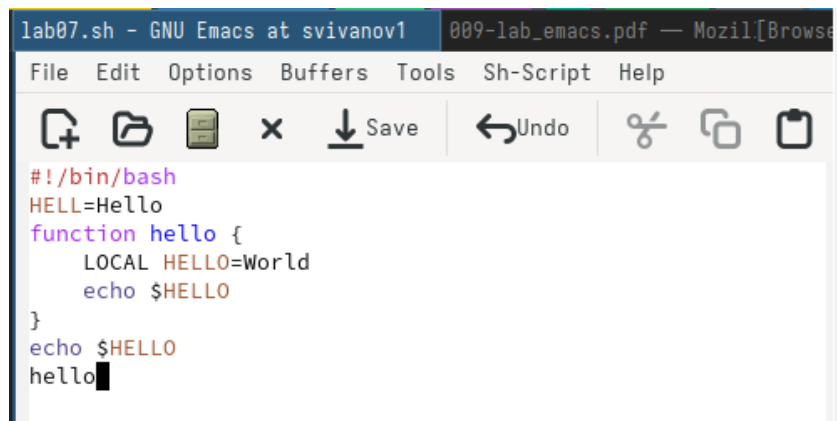


Рис. 3.3: Редактирование файла

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

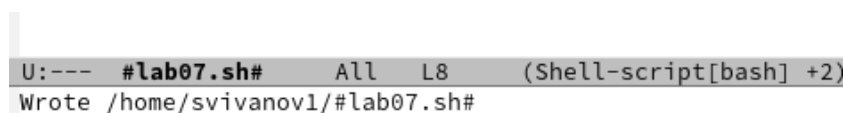


Рис. 3.4: Сохранение изменений в файле

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

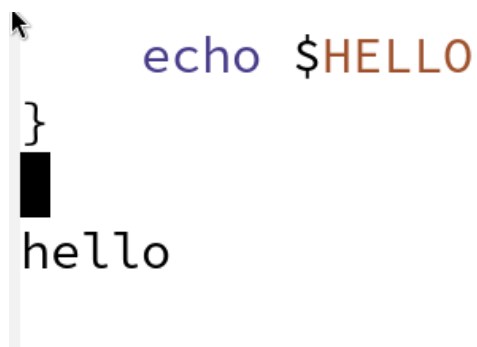


Рис. 3.5: Вырезание строки

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

```

}

hello
echo $HELLO

```

Рис. 3.6: 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).

```

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

hello
echo $HELLO

```

Рис. 3.7: Вырезанная область

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



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

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

Рис. 3.8: Вставка в конец файла

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

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

hello
echo $HELLO
```

U:\*\*\* #lab07.sh# All L9 (Shell-script[bash] +3 Ovwrt)  
Undo

Рис. 3.9: Отмена последнего действия

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

```
}  
hello  
echo $HELLO
```

Рис. 3.10: Курсор в начале строки

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

```
hello  
echo $HELLO
```

Рис. 3.11: Курсор в конце строки

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

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

Рис. 3.12: Начало буфер

Перемещаю курсор в конец файла M->(рис. 13).

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

hello
echo $HELLO
```

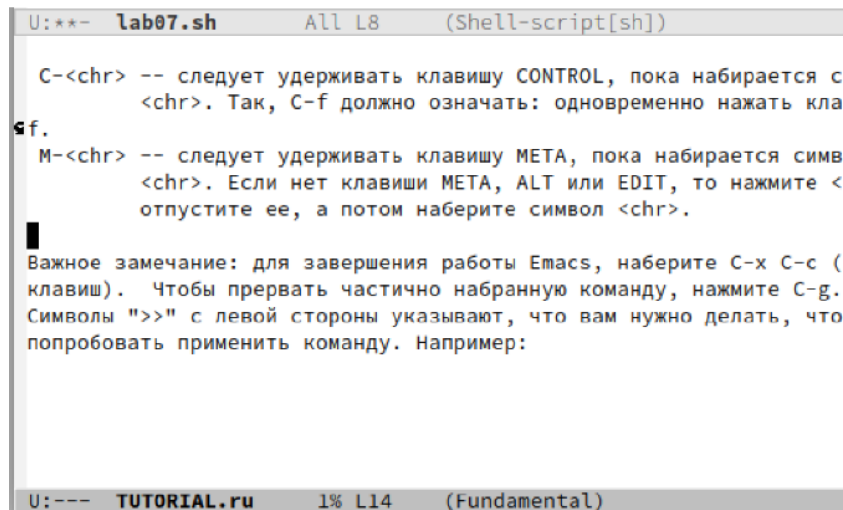
Рис. 3.13: Конец буфера

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

| U:*** #lab07.sh# All L9 (Shell-script[bash] +3 Ovwrt) |                            |      |                                 |
|---|----------------------------|------|---------------------------------|
| CRM   | Buffer                     | Size | Mode File                       |
| [ ]   | * #lab07.sh#               | 102  | Shell-script[ba... ~/#lab07.sh# |
| %   | *GNU Emacs*                | 875  | Fundamental                     |
|   | *scratch*                  | 145  | Lisp Interaction                |
| %*  | *Messages*                 | 2622 | Messages                        |
| %*  | *Async-native-compile-log* | 165  | Fundamental                     |

Рис. 3.14: Список активных буферов

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



U:\*\*\* lab07.sh All L8 (Shell-script[sh])

C-<chr> -- следует удерживать клавишу CONTROL, пока набирается с <chr>. Так, C-f должно означать: одновременно нажать кла f.

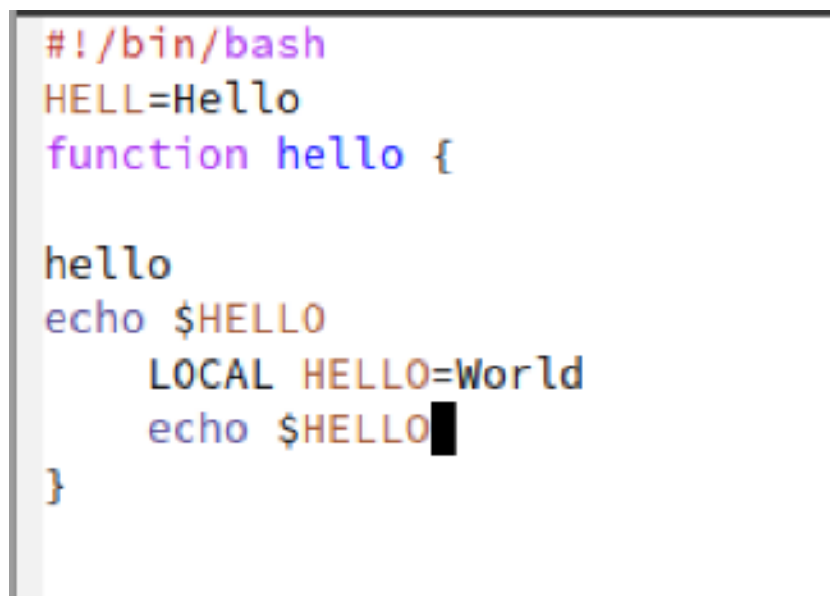
M-<chr> -- следует удерживать клавишу META, пока набирается симв <chr>. Если нет клавиши META, ALT или EDIT, то нажмите < отпустите ее, а потом наберите символ <chr>.

Важное замечание: для завершения работы Emacs, наберите C-x C-c (клавиш). Чтобы прервать частично набранную команду, нажмите C-g. Символы ">>" с левой стороны указывают, что вам нужно делать, что попробовать применить команду. Например:

U:--- TUTORIAL.ru 1% L14 (Fundamental)

Рис. 3.15: Другое окно буфера

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



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

Рис. 3.16: Заккрытие буфера

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

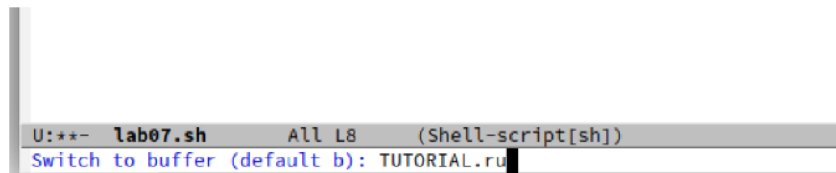


Рис. 3.17: Открытие другого буфера

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

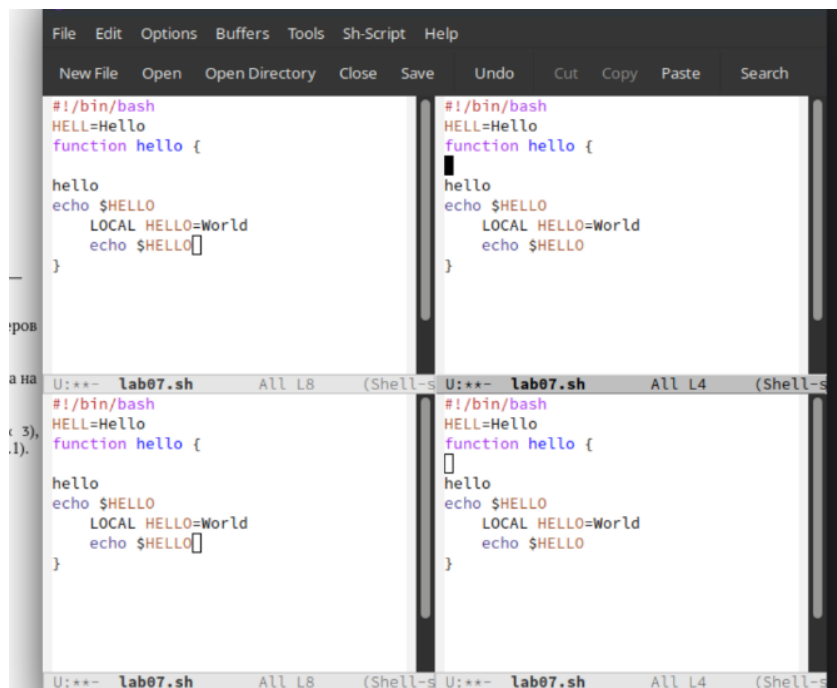


Рис. 3.18: Четыре окна

В каждом из четырех созданных окон с помощью С-х b открыл разные буферы(рис. 19).



Рис. 3.19: Открытие буферов в четырех окнах

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

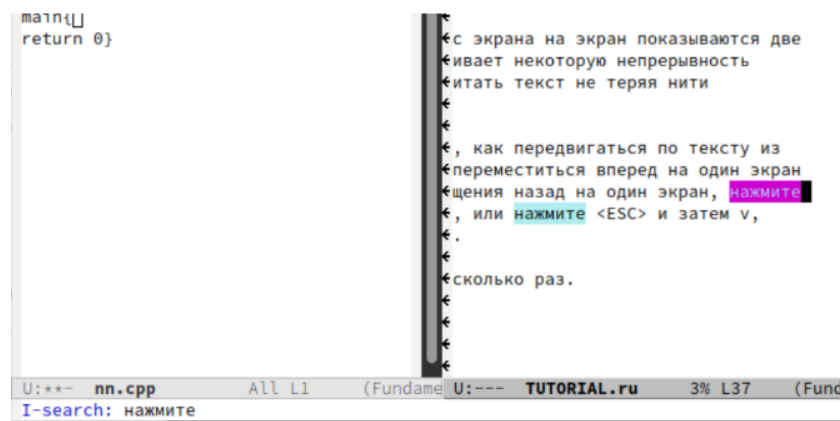


Рис. 3.20: Режим поиска

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

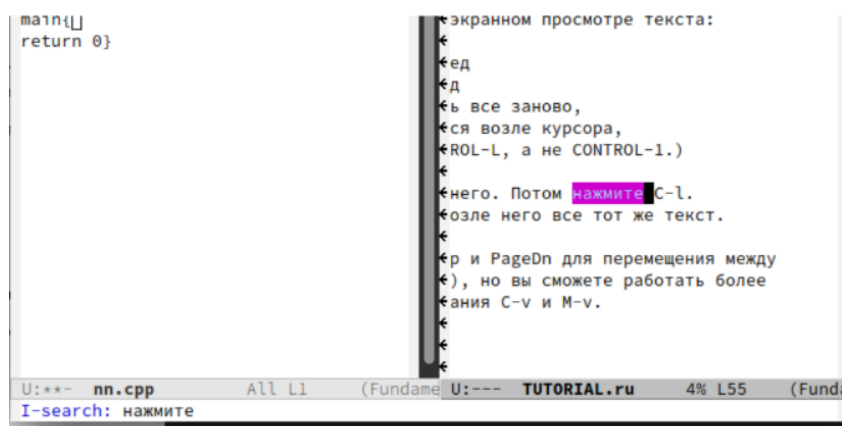


Рис. 3.21: Перемещение по найденным выражениям

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

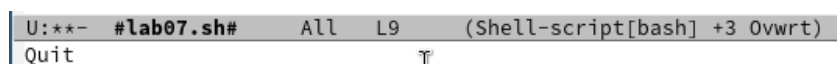


Рис. 3.22: Выход из режима поиска

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

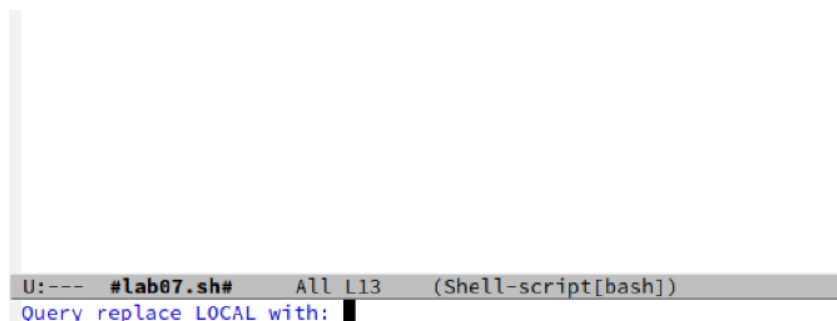


Рис. 3.23: Замена слова

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

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

hello
echo $HELLO
local HELLO= World
local HELLO= World
    echo $HELLO
}
    echo $HELLO
}
```

Рис. 3.24: 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).

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

hello
echo $HELLO
local HELLO= World
local HELLO= World
    echo $HELLO
}
    echo $HELLO
}
```

---

```
U:***- #lab07.sh# All L13 (Shell-script(bash))
7 matches for "local" in buffer: #lab07.sh#
7:local HELLO= World
8:local HELLO= World
```

Рис. 3.25: Режим поиска



## 4 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.

2. 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 работа с Emacs.

3. 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.

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

Да, можно.

5. 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.

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

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

7. Как поделить текущее окно на две части?

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

8. 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.

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

Выполняет функцию стереть, думаю можно переназначить.

10. 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.

## 5 Вывод

В ходе данной лабораторной работы я познакомился с операционной системой Linux, получил практические навыки работы с редактором Emacs.