

1 Запуск Emacs

Для запуска GNU Emacs 30, просто наберите имя программы: `emacs`

2 Выход из Emacs

Приостановить работу Emacs (или свернуть в иконку при работе в X)	<code>C-z</code>
Выйти из Emacs	<code>C-x C-c</code>

3 Файлы

открыть файл в Emacs	<code>C-x C-f</code>
сохранить файл обратно на диск	<code>C-x C-s</code>
сохранить все файлы	<code>C-x s</code>
вставить содержимое другого файла в текущий буфер	<code>C-x i</code>
открыть нужный вам файл вместо текущего	<code>C-x C-v</code>
сохранить буфер в указанном файле	<code>C-x C-w</code>
Переключить состояние буфера “только для чтения”	<code>C-x C-q</code>

4 Получение помощи

Справочная система достаточно проста. Нажмите `C-h` (или `F1`) и следуйте инструкциям. Если вы в первый раз работаете с Emacs, то наберите `C-u C-h t Russian RET` для вызова **учебника**.

Скрыть окно справки	<code>C-x l</code>
Прокрутить окно справки	<code>C-M-v</code>
argoros: показать команды, соответствующие строке	<code>C-h a</code>
показать имя функции, которая запускается клавишей	<code>C-h k</code>
получить информацию о функции	<code>C-h f</code>
получить информацию о режиме	<code>C-h m</code>

5 Исправление ошибок

прервать частично набранную или исполняемую команду	<code>C-g</code>
восстановить файл, потерянный при крахе системы	<code>M-x recover-file</code>
отменить нежелательные изменения	<code>C-x u, C-_ или C-/</code>
восстановить буфер в первоначальное состояние	<code>M-x revert-buffer</code>
перерисовать заполненный мусором экран	<code>C-l</code>

6 Нарастиваемый поиск

поиск вперед	<code>C-s</code>
поиск назад	<code>C-r</code>
поиск по регулярному выражению вперед	<code>C-M-s</code>
поиск по регулярному выражению назад	<code>C-M-r</code>
выбрать предыдущую строку поиска	<code>M-p</code>
выбрать следующую строку поиска	<code>M-n</code>
прекратить нарастиваемый поиск	<code>RET</code>
отменить эффект ввода последнего символа	<code>DEL</code>
прервать текущий поиск	<code>C-g</code>

Для повтора поиска в любом направлении используйте клавиши `C-s` или `C-r`. Если Emacs все еще производит поиск, `C-g` отменит только не выполненную часть.

7 Перемещение

единицы перемещения

символ
слово
строка
начало (или конец) строки
предложение
абзац
страница
выражение (sexp)
функция
переход в начало (или конец) буфера
прокрутка экрана вниз
прокрутка экрана вверх
прокрутка влево
прокрутка вправо
установка текущей строки в центр экрана

назад

C-b
M-b
C-p
C-a
M-a
M-{
C-x [
C-M-b
C-M-a
M-<
C-v
M-v
C-x <
C-x >
C-u C-l

вперед

C-f
M-f
C-n
C-e
M-e
M-}
C-x]
C-M-f
C-M-e
M->

8 Уничтожение и удаление

уничтожаемый объект

символ (удаление, а не уничтожение)
слово
строка (до конца строки)
предложение
выражение
уничтожить **область**
скопировать область в список уничтожений
уничтожить до следующего вхождения символа *char*
вставить последний уничтоженный объект
заменить только что восстановленный текст предшествующим уничтоженным текстом

назад

DEL
M-DEL
M-0 C-k
C-x DEL
M- C-M-k
C-w
M-w
M-z *char*
C-y
M-y

вперед

C-d
M-d
C-k
M-k
C-M-k

9 Пометка

установить пометки
поменять местами курсор и пометку
пометить *arg* **слов**
пометить **параграф**
пометить **страницу**
пометить **выражение**
пометить **функцию**
пометить весь **буфер**

C-@ или C-SPC

C-x C-x
M-@
M-h
C-x C-p
C-M-@
C-M-h
C-x h

10 Замена с запросом

интерактивная замена текстовой строки
с использованием регулярных выражений

В режиме интерактивной замены пользователю доступны следующие команды:

заменить данное соответствие, перейти к следующему

заменить это соответствие, не перемещаться

пропустить соответствие без замены

заменить все оставшиеся соответствия

вернуться к предыдущему соответствию

выйти из режима “замена с запросом”

войти в режим рекурсивного редактирования (для выхода используется C-M-c)

M-%

M-x query-replace-regexp

SPC

,

DEL

!

^

RET

C-r

11 Использование нескольких окон

Команды, приведенные во втором столбце, применяются к другому фрейму (окну используемой оконной системы).

удалить все остальные окна	C-x 1	
разбить окно по горизонтали	C-x 2	C-x 5 2
удалить данное окно	C-x 0	C-x 5 0
разбить окно по вертикали	C-x 3	
выполнить прокрутку в другом окне	C-M-v	
переместить курсор в другое окно	C-x o	C-x 5 o
выбрать буфер в другом окне	C-x 4 b	C-x 5 b
показать буфер в другом окне	C-x 4 C-o	C-x 5 C-o
загрузить файл в другое окно	C-x 4 f	C-x 5 f
открыть файл в режиме чтения в другом окне	C-x 4 r	C-x 5 r
запустить Dired в другом окне	C-x 4 d	C-x 5 d
найти таг в другом окне	C-x 4 .	C-x 5 .
увеличить высоту окна	C-x ^	
сузить текущее окно	C-x {	
расширить текущее окно	C-x }	

12 Форматирование

сделать отступ для текущей строки (в зависимости от режима)	TAB
сделать отступ для области (в зависимости от режима)	C-M-\
сделать отступ для выражения (в зависимости от режима)	C-M-q
сделать отступ к заданной <i>arg</i> колонке	C-x TAB
вставить новую строку после курсора	C-o
сдвинуть остаток строки вертикально вниз	C-M-o
удалить пустые строки вокруг курсора	C-x C-o
объединить строку с предыдущей (со следующими <i>arg</i>)	M-^
удалить пустое пространство вокруг курсора	M-\
поместить только один пробел в позиции курсора	M-SPC
отформатировать текущий параграф	M-q
установить столбец заполнения	C-x f
установить префикс заполнения каждой строки	C-x .
установить начертание (<i>face</i>)	M-g

13 Изменение регистра

сделать буквы слова заглавными	M-u
сделать буквы слова строчными	M-l
сделать первую букву слова заглавной	M-c
сделать буквы области заглавными	C-x C-u
сделать буквы области строчными	C-x C-l

14 Минибуфер

Следующие клавиши (ключи) определены в минибуфере.

дополнить насколько возможно	TAB
дополнить до одного слова	SPC
дополнить и выполнить	RET
показать возможные дополнения	?
выбрать предыдущую строку из истории ввода	M-p
выбрать следующую строку из истории ввода	M-n
поиск в истории по регулярному выражению в обратном направлении	M-r
поиск в истории по регулярному выражению в прямом направлении	M-s
прервать команду	C-g

Наберите C-x ESC ESC для редактирования и повторения последней команды набранной в минибуфере. Наберите F10, чтобы работать с пунктами меню при помощи команд минибуфера.

15 Буфера

выбрать другой буфер
выдать список всех буферов
уничтожить буфер

C-x b
C-x C-b
C-x k

16 Перестановка данных

обменять местами **символы**
обменять местами **слова**
обменять местами **строки**
обменять местами **выражения**

C-t
M-t
C-x C-t
C-M-t

17 Проверка правописания

проверить текущее слово
проверить все слова в области
выполнить проверку всего буфера

M-\$
M-x ispell-region
M-x ispell-buffer

18 Теги

найти определение тега
использовать новый файл с тегами
поиск по шаблону по всей таблице тегов
выполнить query-replace над всеми файлами

M-.
M-x visit-tags-table
M-x tags-search
M-x tags-query-replace

19 Командные процессоры

Выполнить команду командного процессора
выполнить команду командного процессора над областью
пропустить содержимое области через команду командного процессора
запустить командный процессор в окне ***shell***

M-!
M-|
C-u M-|
M-x shell

20 Прямоугольные области

скопировать прямоугольную область в регистр
удалить прямоугольную область
вставить последнюю уничтоженную прямоугольную область
открыть прямоугольную область, сдвигая текст вправо
очистить прямоугольную область
предварить каждую строку строкой текста

C-x r r
C-x r k
C-x r y
C-x r o
C-x r c
C-x r t

21 Сокращения

добавить глобальное сокращение
добавить локальное для режима сокращения
добавить глобальное значение для данного сокращения
добавить специфическое для режима значение данного сокращения
явно вставить значение для сокращения
динамически дополнить предыдущее слово

C-x a g
C-x a l
C-x a i g
C-x a i l
C-x a e
M-/

22 Регулярные выражения

любой одиночный символ (за исключением символа новой строки)
ноль или более повторений
одно или более повторений
ноль или одно повторение
экранировать специальный символ *c*
выбор между альтернативами (“или”)
группировка
текст в *n*-й группе
граница слова
не граница слова
объект

строка
слово
буфер
класс символов

явный набор
символ-буква
символ с синтаксисом *c*

. (dot)
*
+
?
\ *c*
\\
\(... \
\ *n*
\ *b*
\ *B*
**начало соот-
ветствия** **конец соот-
ветствия**
^ \$
\< \>
\‘ \’
**соответ-
ствует им** **соответ-
ствует дру-
гим**
[...] [^...]
\ *w* \ *W*
\ *sc* \ *Sc*

23 Международные наборы символов

выбрать язык
показать все возможные методы ввода
разрешить или запретить метод ввода
установить систему кодирования для следующей команды
показать список всех систем кодирования
выбрать предпочтительную систему кодирования

C-x RET l
M-x list-input-methods
C-\
C-x RET c
M-x list-coding-systems
M-x prefer-coding-system

24 Информация (система Info)

запустить систему доступа к документации
найти указанную функцию или переменную в документации
Перемещение внутри узла (нодами):
прокрутка вперед
прокрутка назад
перейти к началу узла
Перемещение между узлами:
следующий узел
предыдущий узел
перейти к **верхнему** узлу
выбрать пункт меню по имени
выбрать *n*-й пункт меню по номеру (1–9)
перейти по ссылке (возврат с помощью 1)
возвратится к последнему просмотренному узлу
возвратится к узлу-каталогу
Перейти к верхнему узлу файла
перейти к узлу, заданному именем
Другие команды:
запустить **учебник** по системе документации
поиск темы в индексе
поиск в узлах с помощью регулярных выражений
прекратить работу с документацией

C-h i
C-h S

SPC
DEL
. (dot)

n
p
u
m
n
f
l
d
t
g

h
i
s
q

25 Регистры

сохранить область в регистре
вставить содержимое регистра в буфер
сохранить положение курсора в регистре
перейти к положению курсора, сохраненному в регистре

C-x r s
C-x r i
C-x r SPC
C-x r j

26 Клавиатурные макросы

начать определение клавиатурного макроса
закончить определение клавиатурного макроса
выполнить последний определенный макрос
дополнить последний определенный макрос
задать имя для последнего определенного макроса
вставить в буфер определение макроса на языке Lisp

C-x (
C-x)
C-x e
C-u C-x (
M-x name-last-kbd-macro
M-x insert-kbd-macro

27 Команды, связанные с Emacs Lisp

вычислить **выражение** перед курсором
вычислить текущую функцию **defun**
вычислить **область** как выражение
считать и вычислить выражение из минибuffers
загрузить из стандартного системного каталога

C-x C-e
C-M-x
M-x eval-region
M-:
M-x load-library

28 Простая настройка

настроить переменные и оформление (faces)
Создание глобальных привязок клавиш (ключей) на Emacs Lisp (примеры):

```
(global-set-key "\C-cg" 'goto-line)  
(global-set-key "\M-#" 'query-replace-regexp)
```

29 Написание команд

```
(defun <command-name> (<args>)  
  "<documentation>" (interactive "<template>")  
  <body>)
```

Пример:

```
(defun this-line-to-top-of-window (line)  
  "Reposition line point is on to top of window.  
  With ARG, put point on line ARG."  
  (interactive "P")  
  (recenter (if (null line)  
                0  
                (prefix-numeric-value line))))
```

Спецификация `interactive` сообщает о том, что аргумент надо считать интерактивно. Для дополнительной информации наберите `C-h f interactive`.