

Transpose Text

Operation	Keystroke	Function	Note
Transpose (swap) text elements	Emacs supports the following commands that swap a text element with the preceding one. <ul style="list-style-type: none"> • PEL provides extra key bindings, available for convenience and discovery via which-key. The Emacs native keys are shorter. • PEL also provides key bindings for more native Emacs commands. 		
Last updated on:	2025-09-16		
Open this PDF file. See also: ↗ Help/Info	<code><f11> t t <f1></code>	(pel-help-pdf &optional OPEN-WEB-PAGE)	Open the ↗ Transpose local PDF. If the prefix argument (like C-u or M--) is used, then it opens the remote GitHub hosted raw PDF instead. If the pel-flip-help-pdf-arg user-option is set it's the other way around.
Transpose two characters	<ul style="list-style-type: none"> • C-t • <code><f11> t t c</code> 	(transpose-chars ARG)	Interchange characters around point, moving forward one character. <ul style="list-style-type: none"> • With prefix arg ARG, effect is to take character before point and drag it forward past ARG other characters (backward if ARG negative). • If no argument and at end of line, the previous two chars are exchanged.
Transpose two words	<ul style="list-style-type: none"> • M-t • <code><f11> t t w</code> 	(transpose-words ARG)	Interchange words around point, leaving point at end of them. <ul style="list-style-type: none"> • With prefix arg ARG, effect is to take word before or around point and drag it forward past ARG other words (backward if ARG negative). • If ARG is zero, the words around or after point and around or after mark are interchanged. <p>To use: put cursor between 2 words to transpose.</p> <ul style="list-style-type: none"> • After transposition emacs leaves the cursor after the second word. • Emacs transpose words, but not punctuation. The concept of word is, however, flexible because it is controlled by Emacs syntax tables. <p>👉 See Emacs Text - Words section for more information on how to operate on words with Emacs. Also see ↗ Text Modes .</p>
Transpose two sub-words <ul style="list-style-type: none"> • subword-mode must be on • 2 words can be separated by dash, underscore or the 2 parts of a camelCase word. • See: ↗ Text Modes 	<code><f11> t t W</code>	(subword-transpose ARG)	Do the same as 'transpose-words' but on subwords. <ul style="list-style-type: none"> • See the command 'subword-mode' in ↗ Text Modes for a description of subwords. • Optional argument ARG is the same as for 'transpose-words'. <p>👉 Use this transpose words when the subword-mode is active. Useful to transpose words separated by dash, underscores or camelCase words.</p>
Transpose two lines	<ul style="list-style-type: none"> • C-x C-t • <code><f11> t t l</code> 	(transpose-lines ARG)	Exchange current line and previous line, leaving point after both. <ul style="list-style-type: none"> • With argument ARG, takes previous line and moves it past ARG lines. • With argument 0, interchanges line point is in with line mark is in. <p>To use: put cursor to the second line.</p>
Transpose two balanced expressions (sexps)	<ul style="list-style-type: none"> • C-M-t • <code><f11> t t x</code> 	(transpose-sexps ARG)	Transpose 2 balanced expressions (text enclosed in parenthesis, braces, square or angle brackets, quotes, back-quotes and double quotes) of the same of different types. Here they are globally identified as <i>sexp</i> . <ul style="list-style-type: none"> • Unlike 'transpose-words', point must be between the two sexps and not in the middle of a sexp to be transposed. • With non-zero prefix arg ARG, effect is to take the sexp before point and drag it forward past ARG other sexps (backward if ARG is negative). • If ARG is zero, the sexps ending at or after point and at or after mark are interchanged.
Transpose two sentences	<code><f11> t t s</code>	(transpose-sentences ARG)	Interchange the current sentence with the next one. <ul style="list-style-type: none"> • With prefix argument ARG a non-zero integer, moves the current sentence past ARG sentences, leaving point after the current sentence. • If ARG is positive, moves the current sentence forwards, if ARG is negative moves it backwards. If ARG is zero, exchanges the current sentence with the one containing the mark. <p>Note that by default sentences must be separated by end of line or 2 space characters. The number of space characters required can be modified. The PEL package provides the (pel-toggle-sentence-end) command, bound to <code><f11> t m s</code> to toggle the value from 2 to 1. See the Text Modes table.</p> <p>👉 See Emacs Text - Sentences section for more information on how Emacs defines a sentence and the commands dealing with them.</p>
Transpose two paragraphs	<code><f11> t t p</code>	(transpose-paragraphs ARG)	Interchange the current paragraph with the next one. <ul style="list-style-type: none"> • With prefix argument ARG a non-zero integer, moves the current paragraph past ARG paragraphs, leaving point after the current paragraph. • If ARG is positive, moves the current paragraph forwards, if ARG is negative moves it backwards. If ARG is zero, exchanges the current paragraph with the one containing the mark. <p>👉 See Emacs Text - Paragraphs section for more information on how Emacs define a paragraph and the commands dealing with them.</p>
Transpose 2 regions	<code><f11> t t r</code>	(transpose-regions STARTR1 ENDR1 STARTR2 ENDR2 &optional LEAVE-MARKERS)	Transpose region STARTR1 to ENDR1 with STARTR2 to ENDR2. <ul style="list-style-type: none"> • The regions should not be overlapping, because the size of the buffer is never changed in a transposition. • Optional fifth arg LEAVE-MARKERS, if non-nil, means don't update any markers that happen to be located in the regions. • Transposing beyond buffer boundaries is an error. • Interactively, STARTR1 and ENDR1 are point and mark; STARTR2 and ENDR2 are the last two marks pushed to the mark ring; LEAVE-MARKERS is nil. • If a prefix argument N is given, STARTR2 and ENDR2 are the two successive marks N entries back in the mark ring. • A negative prefix argument instead counts forward from the oldest mark in the mark ring. <p>To use interactively:</p> <ol style="list-style-type: none"> 1. First move point at the beginning of the first region. 2. Type C-SPC and move point to the end of the first region. 3. Type C-g 4. Move point to the beginning of the second region. 5. Type C-SPC and move point to the end of the second region. 6. Execute the command.

Transpose — References

Topic & Link	Notes
GNU Emacs Manual: Fixit - Transposing Text	
GNU Emacs Manual: Text - Words	
GNU Emacs Manual: Text - Sentences	
GNU Emacs Manual: Text - Paragraphs	