















Emacs support for C3

Description	Keystroke	Function	Note
<div><div>C3 Support</div><div><div>• File associations</div><div>Last updated on: 2025-12-29</div></div></div>	<div>C3 is a programming language based off the C language. It is relatively new and still evolving.</div> <div>PEL supports the only supported mode for C3: the tree-sitter-based c3-ts-mode provided by the external c3-ts-mode package.</div> <div><div><div>📦</div> Requires the c3-ts-mode file <div>📄</div> PEL installs it in the utils directory when pel-use-c3 user-option is set to t.</div><div>• PEL associates the files with the .c3, .c3i and .c3t file extensions with c3-ts-mode.</div><div><div>⚠️</div> PEL support for C3 requires Emacs >= 30.1 because tree-sitter is required by c3-ts-mode, and PEL only support tree-sitter for Emacs >= 30.1:<div><div>• See Tree Sitter and Tree-sitter.</div><div>• PEL activates Speedbar support for the C3 files when pel-use-speedbar user-option is on (set to t).</div><div>• imenu support provided by c3-ts-mode is available.</div></div></div></div>		
<div>Open this PDF file. See also: Help/Info</div>	<div><f11> SPC M-G <f1></div> <div><f12> <f1></div>	<div>(pel-help-pdf &optional OPEN-WEB-PAGE)</div>	<div>Open the 📄 - C3 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.</div>
<div>Customize PEL C3 support</div>	<div><f11> SPC M-G <f2></div> <div><f12> <f2></div>	<div>(pel-customize-pel &optional OTHER-WINDOW)</div>	<div>Customize PEL C3 support.</div> <div>• If OTHER-WINDOW is non-nil (use C-u), display in another window.</div>
<div>Customize Emacs C3 support</div>	<div><f11> SPC M-G <f3></div> <div><f12> <f3></div>	<div>(pel-customize-library &optional OTHER-WINDOW)</div>	<div>Customize Emacs C3 support: c3-ts.</div> <div>• If OTHER-WINDOW is non-nil (use C-u), display in another window.</div> <div><div>👉</div> Several aspects of c3-ts-mode are controlled by PEL as defined in the user-options accessed by <f12> <f2> and override the c3-ts ones.</div>
<div>Show PEL setup for C3</div>	<div><f11> SPC M-G ?</div> <div><f12> ?</div>	<div>(pel-c3-setup-info & optional APPEND)</div>	<div>Display C3 setup information inside a *pel-c3-info* buffer with buttons providing quick access to the customization buffer of each variable shown. The information shown includes the value and interpretation of: c3-ts-indent-offset, tab-width, activated minor modes</div> <div>To append information in the buffer instead of clearing the previous content type any prefix argument (such as C-u) before the command keystroke.</div>
<div>Set visual rendering of hard tabs for the current buffer</div> <div>See Indentation for more information and commands.</div>	<div><f11> <tab> w</div> <div><f12> M-t</div>	<div>(pel-set-tab-width N)</div>	<div>Change the tab width of the current buffer, only affecting the display rendering of hard tabs inserted in the buffer text. Prompts for a new value in the [2, 8] range.</div> <div>• This modifies a buffer local value of the the tab-width user-option.</div> <div>• The change is temporary and affects the current buffer only.</div> <div>• To change the tab width used for all C3 source code files, change the 'pel-c3-tab-width' user-option variable instead.</div>
Comments	See also: Comments		
<div>Insert, realign, comment/uncomment region</div> <div>With PEL: Comment the current line with M-0 M-;</div>	<div>M-;</div>	<div>(comment-dwim ARG)</div> <div>(pel-comment-dwim ARG)</div>	<div>Insert or realign comment on current line (or region if a region is active).</div> <div>If line/region is already commented, uncomment it.</div> <div>• On a single line, the comment is placed <i>after</i> the code.</div> <div>• C-u M-; executes comment-kill</div> <div>Same as comment-dwim but comments the current line with a numeric ARG or 0.</div>
Select C3 comment style	<f12> <f4> M-;	(pel-select-c3-comment-style)	<div>Select from the following C3 comment styles generated by comment-dwim and pel-comment-dwim commands:</div> <div><div>• //</div><div>• /* */</div><div>• <* *></div></div>
Navigation <div>More in Navigation</div>	<div>• Shift selection is supported by some commands, not all. The following symbols are used to identify whether the command supports shifts selection:</div> <div><div>• <div>⬆️</div> This command supports shift selection in GUI and terminal mode.</div><div>• <div>⬇️</div> This command supports shift selection only in GUI mode.</div><div>• <div>⬇️⬇️</div> This command supports shift selection in GUI mode and also in terminal mode under some conditions (described in the description cell for the command).</div><div>• <div>⬆️⬇️</div> This command does not support shift selection. Sometimes for this you can first set the mark before moving.</div></div> <div>• Pressing the Shift key when using the key binding for commands that do not show any of these 3 arrows have no impact on the shift selection (and may be inappropriate for the command).</div>		
<div>• by defun : C3 definitions <div>🌲</div></div>	<div>The commands move point by C3 definitions: functions, macros, structs, bitstructs, enums, unions, constants, alias, typedef.</div> <div>• The <f6> cursor key mappings use <up> and <down> to move to the beginning of the defun, and <left> and <right> to the end of the defun.</div> <div>• In this context the word <i>defun</i> corresponds to any of the C3 definitions listed above.</div> <div><div>🌲</div> These commands are all enhanced by the use of Tree Sitter .</div>		
<div>Backward to beginning of C3 definition</div> <div><div>🌲⬇️</div><div>⬇️</div></div>	<div>• <f6> <up></div> <div>• C-M-a</div> <div>• C-M-<home></div> <div>• C-[C-a</div> <div>• Esc C-a</div>	<div>(beginning-of-defun &optional ARG)</div>	<div>Move backward to the beginning of a C3 definition.</div> <div>• With ARG, do it that many times. Negative ARG means move forward to the ARGth following beginning of defun.</div> <div><div>⚠️</div> This command moves to the beginning go the next definition of the same nesting level of the current location. It skips the nested definitions.</div>
<div>Forward to end of C3 definition</div> <div><div>🌲⬇️</div><div>⬇️</div></div>	<div>• <f6> <right></div> <div>• C-M-e</div> <div>• C-M-<end></div> <div>• C-[C-e</div> <div>• Esc C-e</div>	<div>(end-of-defun &optional ARG)</div>	<div>Move forward to next end of C3 definition.</div> <div>• With argument, do it that many times. Negative argument -N means move back to Nth preceding end of defun.</div> <div><div>⚠️</div> This command moves to the end of the next top-level function or class. It skips the nested definitions.</div>
Forward to start of next C3 definition <div>🌲⬇️</div>	<f6> <down>	(pel-beginning-of-next-defun &optional SILENT DONT-PUSH_MARK)	<div>Move forward to the beginning of the next C3 definition.</div> <div>• Beeps if does not find beginning of next function unless SILENT is non-nil.</div> <div>• If the beginning of next function is found, push the start location to the mark ring unless DONT-PUSH_MARK is non-nil.<div>• Move back to previous position with M-` or <f6><f6></div></div>
Backward to end of previous C3 definition <div>🌲⬇️</div>	<f6> <left>	(pel-end-of-previous-defun &optional SILENT DONT-PUSH_MARK)	<div>Move backwards to the end of the previous C3 definition.</div> <div>• Beeps if does not find end of previous function unless SILENT is non-nil.</div> <div>• If the end of previous function is found, push the start location to the mark ring unless DONT-PUSH_MARK is non-nil.<div>• Move back to previous position with M-` or <f6><f6></div></div>
<div>• by blocks</div>	<div>Blocks can be: pairs of brackets: {},[],{<,}>,"",‘‘. Blocks using parentheses correspond to Lisp S-Expressions (sexp).</div> <div><div>👉</div> The commands move across C3 blocks but also to the next/previous syntax element.</div>		
<div>block backward</div> <div><div>🌲⬇️</div><div>⬇️</div><div>⬇️</div></div>	<div>• C-M-<left></div> <div>• Esc C-<left> <div>⚠️</div></div> <div>• C-M-b</div> <div>• C-[C-b</div> <div>• Esc C-b</div>	<div>(backward-sexp &optional ARG)</div>	<div>Move backward across one balanced expression (sexp).</div> <div>• With ARG, do it that many times. Negative arg -N means move forward across N balanced expressions. This command assumes point is not in a string or comment.</div> <div>• <div>⚠️</div> With PEL: if you want to use Esc C-<left> binding you must ensure that pel-windmove-on-esc-cursor user option is set to nil.</div> <div>❖ C-M-<left> does not work on Windows, but M-<left> works.</div>

Description	Keystroke	Function	Note
	 Several Linux distros map C-M-<left> to desktop workspace operation. In that case you can either use another key binding or change Linux key binding in Systems->settings->keyboard->shortcuts to prevent it from using that key sequence.		
block forward   	<ul style="list-style-type: none">• C-M-<right>	(forward-sexp &optional ARG)	Move forward across one balanced expression (sexp). <ul style="list-style-type: none">• With ARG, do it that many times. Negative arg -N means move backward across N balanced expressions. This command assumes point is not in a string or comment.•  With PEL: if you want to use Esc C-<right> binding you must ensure that pel-windmove-on-esc-cursor user option is set to nil. ❖ C-M-<right> does not work on Windows, but H-<right> does.
	<ul style="list-style-type: none">• Esc C-<right> 		
	<ul style="list-style-type: none">• C-M-f• C-[C-f• Esc C-f		
	 Several Linux distros map C-M-<right> to desktop workspace operation. In that case you can either use another key binding or change Linux key binding in Systems->settings->keyboard->shortcuts to prevent it from using that key sequence.		
Up/inside sexp hierarchy   	<ul style="list-style-type: none">• C-M-<up>	(backward-up-list &optional ARG ESCAPE-STRINGS NO-SYNTAX-CROSSING)	Move backward out of one level of parentheses. <ul style="list-style-type: none">• This command will also work on other parentheses-like expressions defined by the current language mode. With ARG, do this that many times.• A negative argument means move forward but still to a less deep spot.•  With PEL: if you want to use Esc C-<up> binding you must ensure that pel-windmove-on-esc-cursor user option is set to nil. ❖ C-M-<up> does not work on Windows, but H-<up> does.
	<ul style="list-style-type: none">• Esc C-<up>		
	<ul style="list-style-type: none">• C-M-u• C-[C-u• Esc C-u		
Down/inside sexp/block  	<ul style="list-style-type: none">• C-M-<down>	(down-list &optional ARG)	Move forward down one level of parentheses. <ul style="list-style-type: none">• This command will also work on other parentheses-like expressions defined by the current language mode.• With ARG, do this that many times. A negative argument means move backward but still go down a level.• This command assumes point is not in a string or comment.•  With PEL: To use Esc C-<down> binding you must ensure that pel-windmove-on-esc-cursor user option is set to nil. ❖ C-M-<down> does not work on Windows, but H-<down> does.
	<ul style="list-style-type: none">• Esc C-<down>		
	<ul style="list-style-type: none">• C-M-d• C-[C-d• Esc C-d		
Compilation			
Compile C3 file See 📖 Compilation Mode	<f12> c	(pel-c3-compile)	Compile C3 file, show rrors in compilation-mode buffer.

Emacs & C3— References

Document	Notes		
The C3 Programming Language	<ul style="list-style-type: none"> • C3 home • c3c: C3 Compiler @ Github 	Github repos: <ul style="list-style-type: none"> • Awesome C3 Projects @ Github • c3 vendor libraries @ Github 	
Learning C3	<ul style="list-style-type: none"> • What is C3? 		
C3 blogs	<ul style="list-style-type: none"> • C evolved: The C3 programming Language, by Cristopher Lernö 		
C3 LSP servers	<ul style="list-style-type: none"> • c3-lsp: LSP-server for the C3 language. This must be installed manually. See the installation instructions. 		
Emacs support	<ul style="list-style-type: none"> • c3-ts-mode @ Github : tree-sitter-based major-mode for C3. • tree-sitter-c3 @ Github : tree-sitter language grammar for C3. 		