




🚧 Emacs support for the Algol68 Programming Language 🚧

Description	Keystroke	Function	Note
Algol68 Editing <div> <div>Last updated on:</div> <div>2026-01-20</div> </div>	<div> <div>  PEL supports Algol68 via algol68-mode when pel-use-algol is t. PEL does not support a tree-sitter mode for Algol. PEL also provides: <ul style="list-style-type: none"> 🔗 Speedbar support for Algol files listing functions and some types. 🔗 iMenu is supported by algol68-mode. 🚧 </div> <div> 🚧 PEL supports for the Algol programming language is an early implementation not yet fully documented. </div> </div>		
Open this PDF file. See also: 🔗 Help/Info	<f11> SPC 8 <f1>	(pel-help-pdf &optional OPEN-WEB-PAGE)	Open the PEL - Algol 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.
	<f12> <f1>		
🔗 Customize PEL Algol support	<f11> SPC 8 <f2>	(pel-customize-pel &optional OTHER-WINDOW)	Customize PEL Algol support. <ul style="list-style-type: none"> If OTHER-WINDOW is non-nil (use C-u), display in another window.
	<f12> <f2>		
🔗 Customize Emacs Algol support	<f11> SPC 8 <f3>	(pel-customize-library &optional OTHER-WINDOW)	Customize Emacs Algol support: <ul style="list-style-type: none"> If OTHER-WINDOW is non-nil (use C-u), display in another window.
	<f12> <f3>		
Show PEL setup for Algol68 buffer	<f11> ? /	(pel-mode-setup-info &optional APPEND)	Display Algol setup information inside a "pel-mode-info" buffer with buttons providing quick access to the customization buffer of each variable shown. <ul style="list-style-type: none"> To append information in the buffer instead of clearing the previous content type any prefix argument (such as C-u) before the command keystroke.
Comments			
Toggle display of comments in buffer or active region See also: 🔗 Comments	<f11> ; ;	(hide/show-comments-toggle &optional START END)	Toggle hiding/showing of comments in the active region or whole buffer. <ul style="list-style-type: none"> If the region is active then toggle in the region. Otherwise, in the whole buffer. <div>  This requires the hide-comnt.el package (see 🔗 Comments).  PEL activates it when the pel-use-hide-comnt user option is t. </div>

Emacs & Algol — References

Document	Notes
The Algol Programming Language Why Algol in PEL? I used Algol in 1981 at an EPEL lab while developing a numerical analysis tool for my employer at that time. So it's nice to see some Algol code again.	<ul style="list-style-type: none"> Algol @ Wikipedia Algol68 @ Wikipedia Algol68 Homepage
	<ul style="list-style-type: none"> Introduction to Algol 60 Algol publications @ Github Algol 68 Made Easy A PDF book on software preservation site.
Emacs support:	<ul style="list-style-type: none"> algol68-mode @ sourcehut
Algol compilers	<ul style="list-style-type: none"> Algol68g compiler implementation on Github Algol68 Genie Project Example Algol68 programs @ Algol68 Genie Project GCC SC Approves inclusion of Algol 68 Front End: <ul style="list-style-type: none"> GNU Algol 68 compiler on Algol68 Homepage (under development as January 2026) <ul style="list-style-type: none"> gcc/gcc-a68 @ git repo sourceware
Information about Algol	<ul style="list-style-type: none"> Hello World in Algol68
Discussions about Algol	<ul style="list-style-type: none"> "So, to get back to the point: Go vs Algol-68. TBH, I think the 41-year-old language is richer, clearer and more expressive." @ Reddit
Algol code	<ul style="list-style-type: none"> Example Algol68 programs @ Algol68 Genie Project Repository of Algol68 programs on Github by someone learning Algol