\n	toggle relative number
\d	dispatch and make current file
\w	toggle word wrap
F2	YcmCompleter GoTo
F3	YcmCompleter GoToDeclaration
F4	YcmCompleter GoToDefinition
F5	open nerd tree
F8	toggle ctag bar
tcomment	
gc{motion}	Toggle inline comments
gc[Count]c{motion}	Toggle comment with count and motion
gcc	Toggle comment for the current line
g<{motion}	Uncomment region
g	Uncomment the current line
g	Uncomment the current region as block
g>{motion}	Comment region
g>c	Comment the current line
g>b	Comment the current region as block
gc	Toggle comments (visual mode)
g>	Comment selected text (visual mode)
CTags	
C-]	jump to tag
C-t	ctags jump back from tag
gd	goto local declaration
gD	goto global declaration
g]	list matching tags
gC-]	jump to tag if defined once otherwise list matching tags
:tags	list tag stack
\ut	update ctags
Cscope	
C-\ s	show uses of symbol in quickfix
Find	
\gf (Normal)	(global find) start empty Ag for cpp
\gf (Visual)	(global find) Ag with selection for cpp
\f (Normal)	(find)start empty Ag in current file
\f (Visual)	(find) Ag with selection in current file
Replace	
rx	replace all chars in selection with x
:{range}sort u	sort and remove dups
:{range}sort	sort
:{range}sort!	sort reverse
:{range}sort n	numeric sort
:s/foo/bar/g	Change each 'foo' to 'bar' in the current line.
:%s/foo/bar/gc	Change each 'foo' to 'bar' in all the lines., but ask for confirmation first.
:{range}s/foo/bar/gc	Change each 'foo' to 'bar' in all lines within a visual selection, but ask for confirmation first.
:g/pattern/d	remove every line matching pattern

Multiple cursors	
C-n in Normal	highlights the current word under the cursor in Visual mode/finds next
C-n in Visual	virtual cursor at every line and leaves you in Normal mode.
C-p in Visual	remove the current virtual cursor and go back to the previous virtual cursor location.
C-x in Visual	remove the current virtual cursor and skip to the next virtual cursor location.
:{r}MultipleCursorsFind	regex
Ctrlp	
C-p	CtrlP file
\b	CtrlP buffer
\m	CtrlP mixed
\r	CtrlP most recently used
\t	CtrlP buffer tags
F5	(in CtrlP) purge the cache
C-f/C-b	(in CtrlP) cycle between modes
C-j/C-k	(in CtrlP) up/down
C-d	(in CtrlP) switch to filename only search
C-r	(in CtrlP) switch to regexp mode
Format	
\fx	format XML
\fj	format JSON
:%!xmllint --format -	format xml in whole file
:{range}!xmllint --format -	format xml in selection
:%!python -m json.tool	format json in selection
v	enter Visual Mode
C-v	enter Visual Mode
C-v	in insert mode adds tab char
:retab	fix tabs to spaces
:so ~/.vimrc	Reload vimrc
:! wc %	run external command on current file
:noh	turn off search highlight
vim -d file1 file2	diff 2 files
%	in {range} will use whole file
:set pastetoggle :set paste :set nopaste	turn off paste formatting

tmux	
C-a [	enter text mode, hjkl to move, /? to search, C-ud to page
C-a :	enter command mode
C-a c	create window
C-a .	rename window
C-a -	split horizontal
C-a |	split vertical
C-a 1..9	jump to window
Buffer	
C-w ghjk	move to the left/bottom/top/right window
C-w W	move to next window
:sp	horizontal split
:vsp	vertical split
:b	open buffer with filename
:b	open buffer with number
:bd	close current buffer
:bn	next buffer
:bp	prev buffer
C-6	last buffer
:enew	new buffer, with filename creates
:e	reload buffer, with filename creates
:bufdo e	reload buffer all buffers, runs foreach buffer
A	
\a	switch to the header/cpp file
\as	splits and switches
\av	vertical splits and switches
\at	new tab and switches
\an	cycles through next match
Easy Motion	
\\w	easy motion word forward
\\b	easy motion word back
\\k	easy motion up line
\\j	easy motion down line
Errors	
[q	:cprevious prev error
]q	:cnext next error
[Q	:cfirst first line
]Q	:clast last line
:copen	open Quickfix

:!ctags -R --c++-kinds=+p --fields=+iaSl --extra=+q --links=yes .	update tag file
:!find . -iname *.c -iname *.cc -iname *.hpp -iname *.h -iname *.cpp >.cscopelist.tmp; cscope -q -R -b -i .cscopelist.tmp; rm .cscopelist.tmp	update csope db