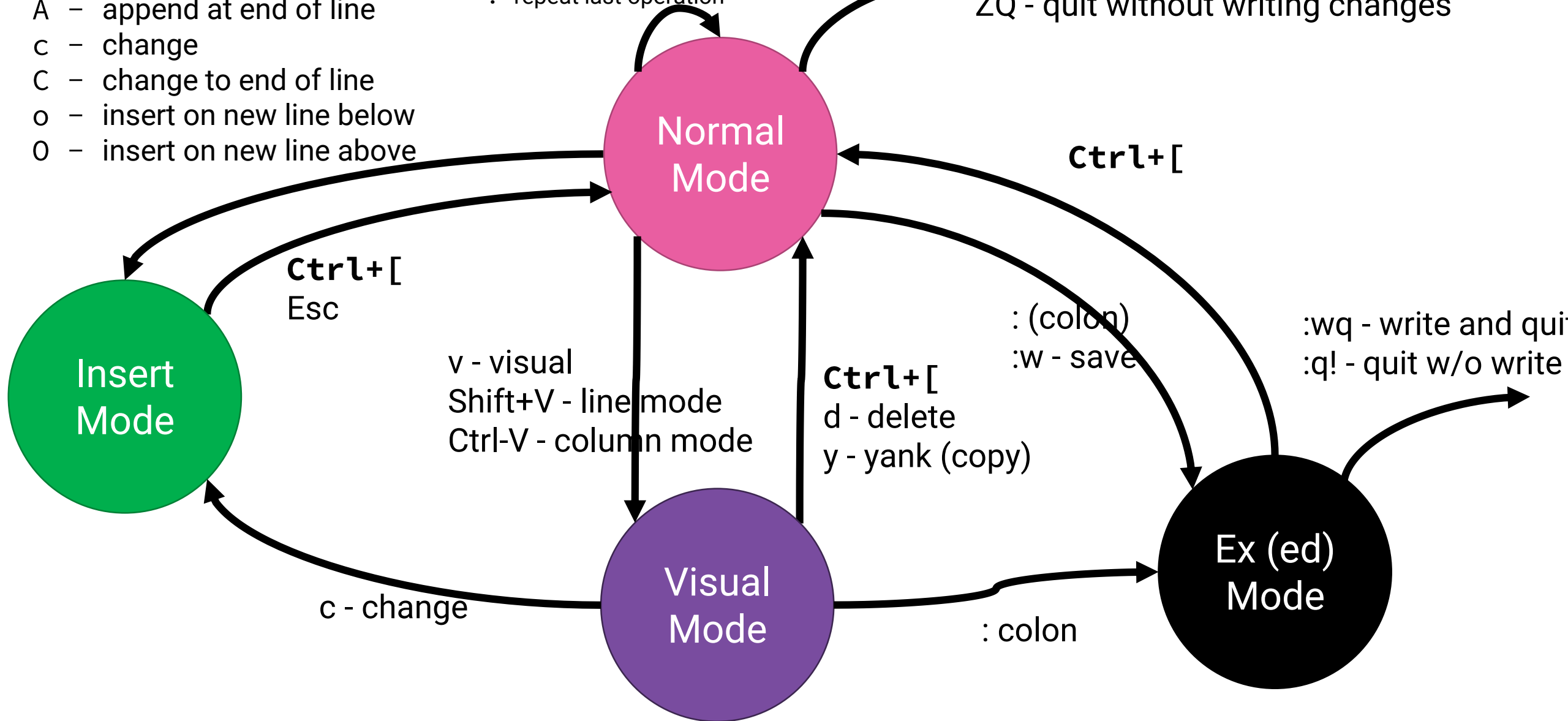


# State Transitions in vim

- i - insert
- I - insert at start of line
- a - append
- A - append at end of line
- c - change
- C - change to end of line
- o - insert on new line below
- O - insert on new line above

- u - undo change
- Ctrl+r - redo
- . - repeat last operation

- ZZ - write and quit (save modifications)
- ZQ - quit without writing changes



# Normal Mode vim Grammar Cheat Sheet

`command` -> **N\_REPEATS?** (cursor\_to | operation | line\_operation | **TO\_INSERT\_MODE** | **MISC**)

cursor\_to -> **LOCATION**

operation -> **VERB** cursor\_to | **VERB** (<sup>inside</sup>'i' | <sup>around</sup>'a') **OBJECT**

line\_operation -> **VERB VERB**

To Insert Mode	Terminal
insert	i
insert at start of line	I
insert new line below	o
insert new line above	O
append after cursor	a
append at end of line	A
change to end of line	C

Save ( <i>write</i> )	:w
Save + Quit	ZZ
Quit no Save	ZQ

Verb	Terminal
change	c
delete	d
yank	y

Object	Terminals
surrounding pair of	" ' } ] )
word	w

Misc	Key
undo / redo	u / ctrl+r
replay last operation	.
paste after / before	p / P
delete a character	x
backspace a character	X
replace a character	r<char>

Location	Terminal
line below	j
line above	k
char left	h
char right	l
first char of line	^
last char of line	\$
next word	w
previous word	b
next end of word	e
find next / prev [c]har in line	f[c] / F[c]
before next / prev [c]har in line	t[c] / T[c]
toggle surrounding (, {, [, ...	%
specific line # of file	[line #]G
last line of file	G
search for "foo" (regexp)	/foo[enter]
next match of last search	n
previous match of last search	N