

In Lisp, 'A is not an atom.

It is a shorthand for

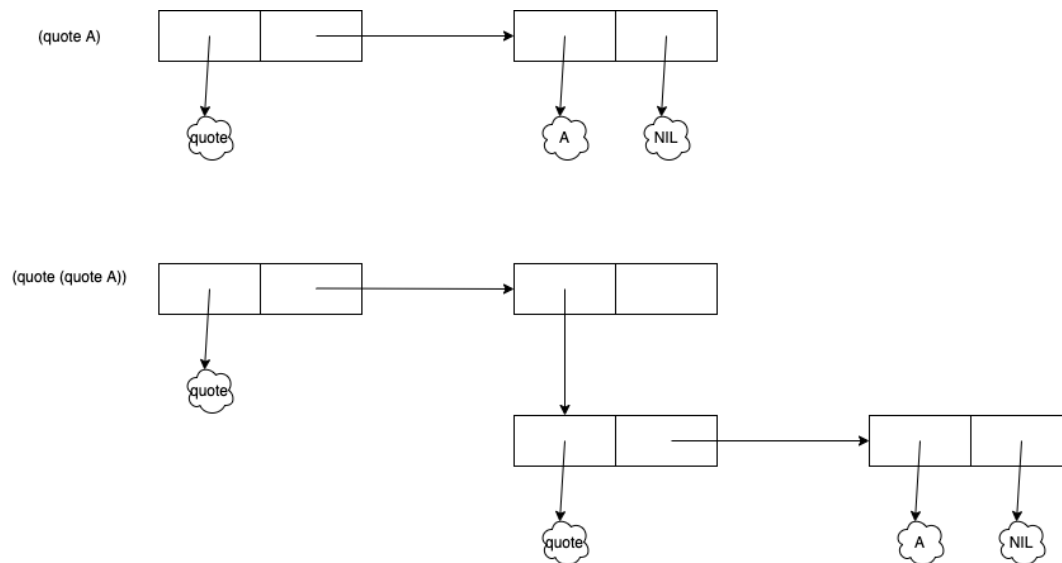
(quote A)

Hence, ''A is shorthand for

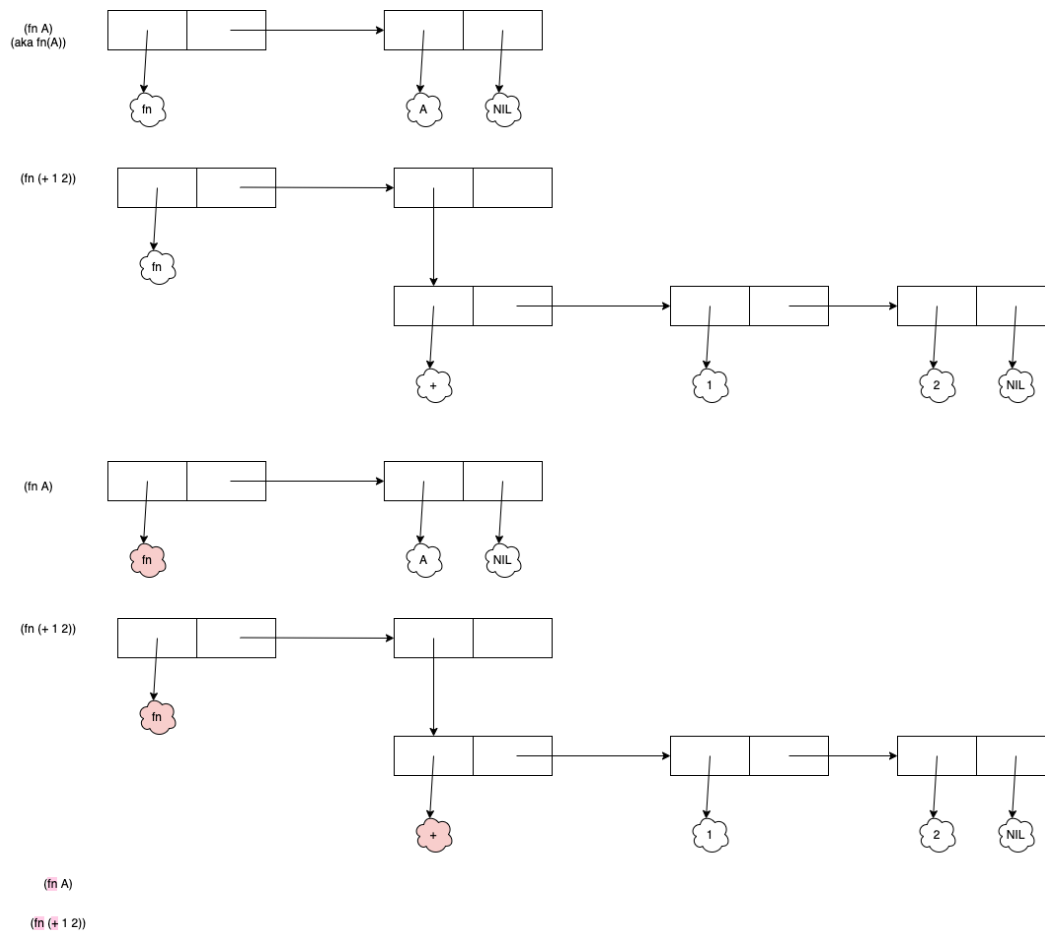
(quote (quote A))

The above are merely lists.

The Lisp compiler (evaluator, eval()) *imparts certain meanings* to lists. The evaluator fetches values from symbols, but stops doing so when it hits *quote*.



[The cloud icons represent *symbols*. Symbols are lists, too, but what is in those lists is implementation dependent. I will skip those details for now.]



In Lisp,

`fn(A)`

is written as

`(fn A)`

and

`fn(1+2)`

is written as

`(fn (+ 1 2))`

In Lisp, when a list is *eval*ed, the first item is considered to be

- a function
- a macro
- something special (a builtin), like *lambda*, *quote*, etc.

I have highlighted, in red, the first items in the above lists.

In Lisp, the *printer* function tries to print lists in a more readable form, for example

- (a b c) is printed as (a b c)
- (quote a) is printed as 'a