

Próximo:[Expansão](#), Acima:[Macros](#) [[Conteúdo](#)][[Índice](#)]

14.1 Um Exemplo Simples de Macro

Suponha que gostaríamos de definir uma construção Lisp para incrementar um valor de variável, muito parecido com o ++operador em C. Gostaríamos de escrever `(inc x)` e ter o efeito de `(setq x (1+ x))`. Aqui está uma definição de macro que faz o trabalho:

```
(defmacro inc (var)
  (lista 'setq var (lista '1+ var)))
```

Quando isso é chamado com `(inc x)`, o argumento *var* é o símbolo *x*— *não* o *valor* de *x*, como seria em uma função. O corpo da macro usa isso para construir a expansão, que é `(setq x (1+ x))`. Uma vez que a definição da macro retorna essa expansão, o Lisp passa a avaliá-la, incrementando assim *x*.

Função: *objeto macrop*

Esse predicado testa se seu argumento é uma macro e retorna em *tcaso* afirmativo, *nil* caso contrário.