

Próximo:[Avaliação durante a expansão](#), Anterior:[Avaliação de argumentos](#), Acima:

[Problemas com macros](#) [Conteúdo][Índice]

14.5.3 Variáveis Locais em Expansões de Macro

Na seção anterior, a definição de `for` foi fixada da seguinte forma para que a expansão avalie os argumentos da macro o número adequado de vezes:

```
(defmacro for (var from init to final do &rest body)
  "Execute um loop for simples: (para i de 1 a 10 faça (imprima i))."
  `(let ((,var ,init)
        (máximo, final))
    (enquanto (<= ,var máx)
      ,@corpo
      (incluindo ,var))))
```

A nova definição de `for` tem um novo problema: ela introduz uma variável local chamada `máx` que o usuário não espera. Isso causa problemas em exemplos como o seguinte:

```
(seja ((máximo 0))
  (para x de 0 a 10 faça
    (deixe ((este (frob x)))
      (se (< max this)
        (setq max this)))))
```

As referências a `max` dentro do corpo do `for`, que deveriam se referir à vinculação do usuário de `max`, realmente acessam a vinculação feita por `for`.

A maneira de corrigir isso é usar um símbolo não interno em vez de `max` (consulte [Criando símbolos](#)). O símbolo não interno pode ser vinculado e referenciado como qualquer outro símbolo, mas como ele é criado por `for`, sabemos que ele não pode aparecer no programa do usuário. Como não é interno, não há como o usuário colocá-lo no programa posteriormente. Ele nunca aparecerá em nenhum lugar, exceto onde colocado por `for`. Aqui está uma definição de `for` que funciona dessa maneira:

```
(defmacro for (var from init to final do &rest body)
  "Execute um loop for simples: (para i de 1 a 10 faça (imprima i))."
  `(let ((tempvar (fazer-símbolo "max")))
    `(let ((,var ,init)
          (tempvar ,final))
      (enquanto (<= ,var ,tempvar)
        ,@corpo
        (inc ,var)))))
```

Isso cria um símbolo não interno nomeado `max` e coloca na expansão em vez do símbolo interno usual `max` que aparece em expressões normalmente.

Próximo:[Avaliação durante a expansão](#), Anterior:[Avaliação de argumentos](#), Acima:

[Problemas com macros](#) [Conteúdo][Índice]

