

Próximo:[Conicionais](#), Acima:[Estruturas de controle](#) [Conteúdo][Índice]

11.1 Sequenciamento

Avaliar formulários na ordem em que aparecem é a maneira mais comum de o controle passar de um formulário para outro. Em alguns contextos, como em um corpo de função, isso acontece automaticamente. Em outros lugares, você deve usar uma construção de estrutura de controle para fazer isso: progn, a construção de controle mais simples de Lisp.

Um prognformulário especial se parece com isso:

```
(prog a b c ...)
```

e diz para executar os formulários *a* , *b* , *c* , e assim por diante, nessa ordem. Esses formulários são chamados de *corpo* do prognformulário. O valor da última forma no corpo torna-se o valor da forma inteira progn. (progn) retorna nil.

Nos primórdios do Lisp, prognera a única maneira de executar duas ou mais formas em sucessão e usar o valor da última delas. Mas os programadores descobriram que muitas vezes precisavam usar a prognno corpo de uma função, onde (na época) apenas uma forma era permitida. Assim, o corpo de uma função foi transformado em um implícito progn: várias formas são permitidas assim como no corpo de uma função real progn. Muitas outras estruturas de controle também contêm um progn. Como resultado, prognnão é usado tanto quanto era há muitos anos. É necessário agora com mais frequência dentro de um unwind-protect, and, or, ou na parte *então* de um if.

Forma Especial: formas de prog ...

Este formulário especial avalia todos os *formulários* , em ordem textual, retornando o resultado do formulário final.

```
(progn (imprima "O primeiro formulário")
      (imprima "A segunda forma")
      (imprima "O terceiro formulário"))
     -| "A primeira forma"
     -| "A segunda forma"
     -| "A terceira forma"
⇒ "A terceira forma"
```

Duas outras construções também avaliam uma série de formulários, mas retornam valores diferentes:

Formulário especial: formulários prog1 form1...

Este formulário especial avalia *form1* e todos os *formulários* , em ordem textual, retornando o resultado de *form1* .

```
(prog1 (imprimir "O primeiro formulário")
      (imprima "A segunda forma")
      (imprima "O terceiro formulário"))
     -| "A primeira forma"
     -| "A segunda forma"
```

```
- | "A terceira forma"
⇒ "A primeira forma"
```

Aqui está uma maneira de remover o primeiro elemento de uma lista na variável `xe`, em seguida, retornar o valor desse elemento anterior:

```
(prog1 (carro x) (setq x (cdr x)))
```

Formulário especial: formulários `prog2 form1 form2...`

Esse formulário especial avalia `form1` , `form2` e todos os *formulários* a seguir , em ordem textual, retornando o resultado de `form2` .

```
(prog2 (imprimir "O primeiro formulário")
      (imprima "A segunda forma")
      (imprima "O terceiro formulário"))
- | "A primeira forma"
- | "A segunda forma"
- | "A terceira forma"
⇒ "A segunda forma"
```

Próximo:[Condicionais](#), Acima:[Estruturas de controle](#) [[Conteúdo](#)][[Índice](#)]