

Próximo:[Funções vetoriais](#), Anterior:[Funções de matriz](#), Acima:[Vetores de matrizes de sequências](#) [[Conteúdo](#)][[Índice](#)]

6.4 Vetores

Um *vetor* é uma matriz de uso geral cujos elementos podem ser quaisquer objetos Lisp. (Em contraste, os elementos de uma string só podem ser caracteres. Veja [Strings e Characters](#) .) Vetores são usados no Emacs para muitos propósitos: como sequências de teclas (veja [Key Sequences](#)), como tabelas de busca de símbolos (veja [Criando Símbolos](#)), como parte da representação de uma função compilada por byte (consulte [Byte Compilation](#)) e muito mais.

Como outras matrizes, os vetores usam indexação de origem zero: o primeiro elemento tem índice 0.

Os vetores são impressos com colchetes ao redor dos elementos. Assim, um vetor cujos elementos são os símbolos a, be aé impresso como [a b a]. Você pode escrever vetores da mesma maneira na entrada Lisp.

Um vetor, como uma string ou um número, é considerado uma constante para avaliação: o resultado da avaliação é o mesmo vetor. Isso não avalia nem examina os elementos do vetor. Consulte [Formulários de autoavaliação](#) . Vetores escritos com colchetes não devem ser modificados por meio a set de outras operações destrutivas. Veja [Mutabilidade](#) .

Aqui estão alguns exemplos que ilustram esses princípios:

```
(setq avector [1 dois '(três) "quatro" [cinco]])
⇒ [1 dois '(três) "quatro" [cinco]]
(vetor eval)
⇒ [1 dois '(três) "quatro" [cinco]]
(vetor eq (vetor eval))
⇒ t
```