

Próximo:[Tipo de string](#), Anterior:[Contras Tipo de Célula](#), Acima:[Tipos de programação](#) [[Conteúdo](#)][[Índice](#)]

2.4.7 Tipo de Matriz

Um *array* é composto por um número arbitrário de slots para armazenar ou referir-se a outros objetos Lisp, dispostos em um bloco contíguo de memória. Acessar qualquer elemento de uma matriz leva aproximadamente a mesma quantidade de tempo. Em contraste, acessar um elemento de uma lista requer tempo proporcional à posição do elemento na lista. (Os elementos no final de uma lista demoram mais para serem acessados do que os elementos no início de uma lista.)

O Emacs define quatro tipos de array: strings, vetores, bool-vectors e char-tables.

Uma string é uma matriz de caracteres e um vetor é uma matriz de objetos arbitrários. Um vetor bool pode conter apenas `nil`. Esses tipos de array podem ter qualquer tamanho até o maior `fixnum`, sujeito aos limites da arquitetura do sistema e à memória disponível. Tabelas de char são matrizes esparsas indexadas por qualquer código de caractere válido; eles podem conter objetos arbitrários.

O primeiro elemento de uma matriz tem índice zero, o segundo elemento tem índice 1 e assim por diante. Isso é chamado de indexação *de origem zero*. Por exemplo, uma matriz de quatro elementos tem índices 0, 1, 2 e 3. O maior valor de índice possível é um a menos que o comprimento da matriz. Uma vez que uma matriz é criada, seu comprimento é fixo.

Todos os arrays Emacs Lisp são unidimensionais. (A maioria das outras linguagens de programação suportam arrays multidimensionais, mas não são essenciais; você pode obter o mesmo efeito com arrays unidimensionais aninhados.) Cada tipo de array tem sua própria sintaxe de leitura; consulte as seções a seguir para obter detalhes.

O tipo de matriz é um subconjunto do tipo de sequência e contém o tipo de string, o tipo de vetor, o tipo bool-vector e o tipo char-table.

Próximo:[Tipo de string](#), Anterior:[Contras Tipo de Célula](#), Acima:[Tipos de programação](#) [[Conteúdo](#)][[Índice](#)]