

Anterior:[Bool-Vetores](#), Acima:[Vetores de matrizes de sequências](#) [Conteúdo][Índice]

## 6.8 Gerenciando um Anel de Objetos de Tamanho Fixo

Um *anel* é uma estrutura de dados de tamanho fixo que suporta inserção, exclusão, rotação e referência e travessia indexada por módulo. Uma estrutura de dados em anel eficiente é implementada pelo `ring` pacote. Ele fornece as funções listadas nesta seção.

Observe que vários anéis no Emacs, como o kill ring e o mark ring, na verdade são implementados como listas simples, *não* usando o `ring` pacote; portanto, as seguintes funções não funcionarão neles.

### Função: *tamanho do anel*

Isso retorna um novo anel capaz de conter objetos de *tamanho*. *tamanho* deve ser um número inteiro.

### Função: *objeto ring-p*

Isso retorna tse o *objeto* for um anel, nil caso contrário.

### Função: *anel do tamanho do anel*

Isso retorna a capacidade máxima do *anel*.

### Função: *anel de comprimento do anel*

Isso retorna o número de objetos que o *anel* contém atualmente. O valor nunca excederá o retornado por `ring-size`.

### Função: *anel de elementos de anel*

Isso retorna uma lista dos objetos em *ring*, em ordem, os mais novos primeiro.

### Função: *anel de cópia de anel*

Isso retorna um novo anel que é uma cópia de *ring*. O novo anel contém os mesmos eqobjetos () do *anel*.

### Função: *anel vazio-p anel*

Isso retorna tse o *anel* estiver vazio, nil caso contrário.

O elemento mais novo no anel sempre tem índice 0. Índices mais altos correspondem a elementos mais antigos. Os índices são calculados módulo o comprimento do anel. O índice -1 corresponde ao elemento mais antigo, -2 ao próximo mais antigo e assim por diante.

### Função: *índice de anel de referência de anel*

Isso retorna o objeto em *anel* encontrado no índice *index*. *índice* pode ser negativo ou maior que o comprimento do anel. Se o *anel* estiver vazio, `ring-ref` finaliza um erro.

### Função: *objeto de anel de inserção de anel*

Isso insere *object* em *ring*, tornando-o o elemento mais novo e retorna *object*.

Se o anel estiver cheio, a inserção remove o elemento mais antigo para dar espaço ao novo elemento.

### Função: anel de remoção do *anel* e *índice opcional*

Remova um objeto de *ring* e devolva esse objeto. O *índice* do argumento especifica qual item remover; se for *nil*, isso significa remover o item mais antigo. Se o *anel* estiver vazio, *ring-remove!* sinaliza um erro.

### Função: *object* de anel de inserção de anel no início

Isso insere *object* em *ring*, tratando-o como o elemento mais antigo. O valor de retorno não é significativo.

Se o anel estiver cheio, esta função remove o elemento mais recente para dar espaço ao elemento inserido.

### Função: tamanho do anel de redimensionamento do *anel*

Defina o tamanho do *anel* para o *tamanho*. Se o novo tamanho for menor, os itens mais antigos do anel serão descartados.

Se você tomar cuidado para não exceder o tamanho do anel, poderá usá-lo como uma fila de primeiro a entrar, primeiro a sair. Por exemplo:

```
(deixe ((fifo (faça-anel 5)))
  (mapc (lambda (obj) (inserir anel fifo obj))
    '(0 um "dois"))
  (listar (remover anel fifo) t
    (remove o anel fifo) t
    (remove o anel fifo)))
  ⇒ (0 t um t "dois"))
```

Anterior:[Bool-Vetores](#), Acima:[Vetores de matrizes de sequências](#) [Conteúdo][Índice]