

Próximo:[Tipo de ponto flutuante](#), Acima:[Tipos de programação](#) [[Conteúdo](#)][[Índice](#)]

### 2.4.1 Tipo Inteiro

Sob o capô, existem dois tipos de inteiros – inteiros pequenos, chamados *fixnums*, e inteiros grandes, chamados *bignums*.

O intervalo de valores para um número fixo depende da máquina. O intervalo mínimo é -536.870.912 a 536.870.911 (30 bits; ou seja,  $-2^{29}$  a  $2^{29} - 1$ ), mas muitas máquinas oferecem um intervalo mais amplo.

Bignums pode ter precisão arbitrária. As operações que estouram um fixnum retornarão um bignum.

Todos os números podem ser comparados com `eq` ou `=`; fixnums também pode ser comparado com `eq`. Para testar se um inteiro é um fixnum ou bignum, você pode compará-lo com `most-negative-fixnum` and `most-positive-fixnum`, ou pode usar os predicados de conveniência `fixnump` e `bignump` em qualquer objeto.

A sintaxe de leitura para inteiros é uma sequência de dígitos (base dez) com um sinal opcional no início e um ponto final opcional. A representação impressa produzida pelo interpretador Lisp nunca tem um 'principal'+ou final'.'.

```
-1; 0 inteiro -1.  
1; 0 inteiro 1.  
1. ; Também o inteiro 1.  
+1 ; Também o inteiro 1.
```

Consulte [Números](#), para obter mais informações.