



A linguagem LISP

Feito por: Higor Alves



Universidade Federal
de São João del-Rei

- História
- Aplicabilidade
- Características Técnicas
- Implementação das Listas
- Exemplos de Funções
- Macros
- Uso de LISP

- Seu criador foi John McCarthy
- Criada no ano de 1956
- A primeira implementação da linguagem se deu em 1958
- Houveram vários dialetos na linguagem conforme o decorrer do tempo
- Primeiras implementações foram realizadas nos IBM 704 e 7090
- Passaram depois a serem implementadas nos DEC PDP-6 e PDP-10
- Feita a mudança por possuírem palavras de 36 bits endereços de 18 bits

- Ferramenta rápida e altamente personalizável para fazer coisas do dia a dia
- Aplicações grandes, complexas e críticas as quais seriam impossíveis desenvolver em outra linguagem
- Prototipação rápida e Rapid Application Development (RAD)
- Aplicações de alta disponibilidade, principalmente aquelas que necessitam de mudanças após a etapa inicial

- Tipos de dados: átomo e a lista.
- Fraca Tipagem
- Funções de ordem elevada
- Avaliação Ociosa
- Concorrência (multitarefa)
- Um alto nível de abstração, especialmente quando as funções são utilizadas, suprimindo muitos detalhes da programação e minimizando a probabilidade da ocorrência de muitas classes de erros;

- A não dependência das operações de atribuição permite aos programas avaliações nas mais diferentes ordens. Esta característica de avaliação independente da ordem torna as linguagens funcionais as mais indicadas para a programação de computadores maciçamente paralelos;
- A ausência de operações de atribuição torna os programas funcionais muito mais simples para provas e análises matemáticas do que os programas procedurais.

E como desvantagem, destacamos:

- Uma menor eficiência para resolver problemas que envolvam muitas variáveis (ex. contas de banco) ou muitas atividades seqüenciais são muitas vezes mais fáceis de se trabalhar com programas procedurais ou programas orientados a objeto.

```
[  .  ]  
 |  |  
 |  +----> ponteiro para a restante lista (quando for o último, aponta para nil)  
+-----> ponteiro para o conteúdo do elemento
```

```
[  .  ] +-> [  .  ] +-> [  .  ] +-> [  .  ] +-> [  .  ]  
 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  
 |  +---+  |  +---+  |  +---+  |  +---+  |  +---+  |  +---> nil  
esta      lista contém 5 átomos
```

Exemplo de Funções

(**quote** expressão)

Retorna a expressão diretamente, sem tentar qualquer forma da avaliação. Ex: (quote jose) retorna jose, e (quote (jose silva)) retorna (jose silva).

'expressão

Significa o mesmo que (**quote** expressão). Ex: 'jose retorna jose, e '(jose silva) retorna (jose silva).

(**eval** expressão)

força a avaliar a expressão. Ex: Embora '(+ 3 4) simplesmente retorna (+ 3 4), (eval '(+ 3 4)) força a avaliar o (+ 3 4) e portanto retorna 7.

(**car** lista)

Retorna o primeiro elemento da lista. Ex: (car '(jose silva)) retorna jose. Entre os vários dialetos de Lisp, há alguns (por exemplo, [ISLISP](#)) que permitem o nome **first** como alternativa para **car**.

(**cdr** lista)

Retorna a lista sem o primeiro elemento. Ex: (cdr '(jose da silva)) retorna (da silva). Há dialetos que usam o nome **rest** como alternativa para **cdr**.

(**cons** atomo lista)

Adiciona átomo ao início da lista. Ex: (cons 'jose '(da silva)) retorna (jose da silva).

Funções matemáticas:

O grande diferencial de Lisp são as [macros](#). As macros são completamente diferentes das que se encontram em C, pois estas somente fazem substituição de texto, enquanto que em Lisp as macros são programas que geram programas.

Lisp foi utilizado para desenvolver o primeiro sistema computacional de [matemática simbólica](#), o [Macsyma](#).

A ITA software desenvolveu um sistema de reserva de passagens chamado Orbitz em LISP

O LISP foi utilizado pelo [Paul Graham](#) para desenvolver o sistema de [e-commerce](#) da Viaweb