Paradigma Lógica

**A ORIGEM:**

O paradigma lógico emergiu na década de 70, sendo fundamentalmente distinto dos outros paradigmas. Inicialmente foi proposto pela primeira vez na programação, por John McCarthy em 1958. O paradigma lógico se baseia na expressão de programas por meio de lógica simbólica e na utilização de um processo de inferência lógica para produzir resultados. Este paradigma é amplamente aplicado nas áreas de inteligência artificial, processamento de linguagem natural, desenvolvimento de sistemas especialistas e demonstração de teoremas.

**SUA EVOLUÇÃO E USO:**

Usando-se da lógica matemática, John McCarthy em 1958 foi o primeiro a publicar uma proposta de uso da lógica matemática para programação.

A primeira linguagem de programação lógica foi Planner, usando estruturas de controle de backtracking, de tal forma que apenas um único caminho computacional tinha que ser armazenado por vez.

O Prolog foi desenvolvido como uma simplificação do Planner, desenvolvida em 1972 por Alain Colmerauer. Ela veio de uma colaboração entre Colmerauer em Marselha e Robert Kowalski em Edinburgo. Desde então tem sido utilizada para aplicações de computação simbólica, como banco de dados relacionais, compreensão de linguagens naturais (português, inglês etc.), automação de projetos, análise de estruturas bioquímicas e sistemas especialistas.

**LINGUAGENS ESPECÍFICAS PARA ESSE PARADIGMA:**

Java, Python, PHP, JavaScript, Prolog, Datalog, Mercury, Alice.

**LINGUAGENS MULTIPARADIGMA QUE INCLUI ESSE PARADIGMA:**

C++, Groovy, Oz, Ruby, Scala, Swift, Lua, Python.

**Um trecho de código que exemplifica o uso desse paradigma (Exemplo em Python):**

from pyDatalog import pyDatalog

# Define alguns fatos sobre a relação de parentesco

pyDatalog.create\_terms('pai, filho, avo, X, Y, Z')

pai('João', 'Pedro')

pai('Pedro', 'Ana')

pai('Pedro', 'Bruno')

pai('Carlos', 'João')

# Define uma regra para a relação de avô

avo(X, Y) <= pai(X, Z) & pai(Z, Y)

# Faz algumas consultas sobre a relação de parentesco

print(pai(X, 'Ana')) # Quem é o pai de Ana?

print(filho('Pedro', X)) # Quem são os filhos de Pedro?

print(avo('Carlos', X)) # Quem são os netos de Carlos?

**Comparativo entre este paradigma e um paradigma imperativo:**

- Variáveis imperativas: armazenam e modificam o estado do programa;

- Variáveis lógicas: podem ser unificadas com valores;

- Estruturas de controle imperativas: laços, condicionais e funções;

- Estruturas de controle lógicas: cláusulas, predicados e consultas;

- Paradigma imperativo: mais próximo dos computadores;

- Paradigma lógico: mais próximo dos humanos;

- Problemas imperativos: algoritmos, cálculos e manipulação de dados;

- Problemas lógicos: inteligência artificial, bancos de dados e linguagens naturais.

**Referencias:**

https://blog.geekhunter.com.br/quais-sao-os-paradigmas-de-programacao/#Paradigma\_de\_Logica\_de\_Programacao

https://leandromoh.gitbooks.io/tcc-paradigmas-de-programacao/content/6\_paradigma\_logico/index.html

https://pt.wikipedia.org/wiki/Programação\_lógica

https://www.trabalhosgratuitos.com/Exatas/Engenharia/Paradigma-Logico-E-Funcional-221295.html

https://guia.dev/pt/pillars/languages-and-tools/programming-paradigms.html

https://pt.linkedin.com/pulse/paradigmas-de-programação-franciele-sena

<https://www.dio.me/articles/paradigmas-de-programacao-F19UUK>