



PARADIGMAS E LINGUAGENS DE PROGRAMAÇÃO

ENGENHARIA DA COMPUTAÇÃO – UFC/SOBRAL

Prof. Danilo Alves

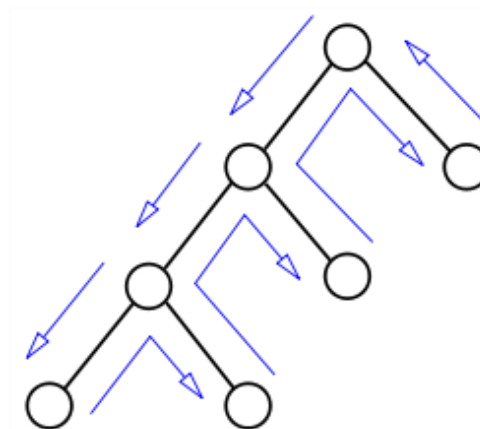
`danilo.alves@alu.ufc.br`

INTRODUÇÃO



UNIVERSIDADE
FEDERAL DO CEARÁ

- Lógica matemática
- John McCarthy (1958) utilizou para programação
- A primeira foi a Planner
 - Usava estruturas de controle de backtracking
- Prolog



\sim	Não
\wedge	e
\vee	ou
\longrightarrow	se... então
\longleftrightarrow	se e somente se
$ $	tal que
\Rightarrow	implica
\Leftrightarrow	equivalente
\exists	existe
$\exists $	existe um e somente um
\forall	qualquer que seja

INTRODUÇÃO



UNIVERSIDADE
FEDERAL DO CEARÁ

- *"programas para manipular com sentenças instrumentais comuns apropriadas à linguagem formal (muito provavelmente uma parte do cálculo de predicado)"*
- O programa básico forma conclusões baseadas em premissas
- Realizar alguma função ou um esquema logico por meio de parâmetros e metas
- Estilo da lógica matemática para computadores
- A lógica na matemática e filosofia



- A lógica é uma ferramenta eficaz para desenvolvimento de teorias
- Problemas são expressos como teorias
- O que é preciso para uma nova hipótese está certa?

Se a nova hipótese é consistente com uma teoria existente ou se é consequência dela.

- Lógica proporciona uma maneira de demonstrar se uma questão é verdadeira ou falsa

INTRODUÇÃO



UNIVERSIDADE
FEDERAL DO CEARÁ

- Linguagens que utilizam o paradigma lógico são
 - Classificadas com Declarativas
 - Descrevem o problema e não a solução
 - Descrevem O QUE deve ser feito e não COMO deve ser feito
 - Conceito de Mundo Fechado
 - Descreve o problema através de FATOS e REGRAS
 - Realiza-se CONSULTAS que deverão ser respondidas pelo interpretador avaliando-se os fatos e regras descritos
 - Tudo o que o interpretador não souber terá uma resposta negativa

PROGRAMAÇÃO DECLARATIVA



UNIVERSIDADE
FEDERAL DO CEARÁ

- Declarativo
 - Modelo de computação baseado em um sistema onde as relações são especificadas diretamente em termos da entrada
 - Atribuição não destrutiva
 - A ordem de execução não importa (não tem efeitos colaterais)
 - O programador não é responsável pelo controle
 - Exemplos: SQL, Prolog, Haskell

PROGRAMAÇÃO LÓGICA



UNIVERSIDADE
FEDERAL DO CEARÁ

- Principal linguagem de programação do paradigma de programação lógico → PROLOG
- Prolog = Programming in Logic
 - Linguagem de programação utilizada para resolver problemas envolvendo objetos e relações entre objetos
 - Ex: Expressar em Prolog o fato “João gosta de Maria”:
 - objetos: “Maria” e “João”
 - relacionamento: “gosta de”, ou simplificando, “gosta”.
 - Em Prolog: `gosta(joao,maria)`.

PARADIGMA LÓGICO



UNIVERSIDADE
FEDERAL DO CEARÁ

- Sua base lógica num subconjunto da logica (cláusulas de Horn)
- Para que expressões ou regras sejam verdadeiras é necessário que todas as regras que a compõem sejam verdadeiras
- Sua execução é principalmente recursiva
- Enfatiza a descrição declarativa de um problema, ao invés da decomposição do problema em uma implementação algorítmica



- Algoritmo é constituído por dois elementos disjuntos: a lógica e o controle
 - Lógica corresponde à definição do que deve ser solucionado
 - Controle estabelece como a solução pode ser obtida
- O programador descreve o componente lógico de um algoritmo, e o controle da execução é exercido pelo sistema de programação em lógica utilizado
- Tarefa de *especificação*

PROGRAMA LÓGICO



UNIVERSIDADE
FEDERAL DO CEARÁ

- Representação de determinado problema através de um conjunto finito de sentenças lógicas
- Não é a descrição de um procedimento para se obter a solução
 - Diferencia de Pascal e C, por exemplo
- Pode também ser visto alternativamente como uma base de dados
- A diferença é que bases de dados representam *fatos* e programas são *regras*
 - "Oscar é um avestruz" → Base
 - "Todo avestruz é um pássaro" → Regra



- Nenhuma linguagem de programação em lógica pode explorar totalmente o potencial da lógica matemática
- Pois o formalismo matemático não é implementável
- Exemplo:
- O Último Teorema de Fermat afirma que não existe nenhum conjunto de inteiros positivos x , y , z e n com n maior que 2 que satisfaça a seguinte equação:

$$z^n = x^n + y^n$$

LÓGICO VS OUTROS PARADIGMAS



UNIVERSIDADE
FEDERAL DO CEARÁ



Lógica



Imperativa



Funcional

PROGRAMAÇÃO IMPERATIVA VS LÓGICA



UNIVERSIDADE
FEDERAL DO CEARÁ

- IMPERATIVA:
 - Programas: mapeamento de entradas em saídas
 - Modelo computacional: von Neumann (variáveis, atribuição, comandos, ...)
 - Estilo procedural: "como fazer"
 - Programação: instruções que executam comandos
 - Computação: ações que alteram estado

PROGRAMAÇÃO IMPERATIVA VS LÓGICA



UNIVERSIDADE
FEDERAL DO CEARÁ

■ LÓGICA:

- Programas: conhecimentos sobre um problema e não uma sequência de passos
- Modelo computacional: lógica matemática
- Estilo declarativo: "o que fazer"
- Programação: simbólica (não numérica)
- Computação: se uma consulta pode ser deduzida de relações do programa

PROGRAMAÇÃO FUNCIONAL VS. LÓGICA



UNIVERSIDADE
FEDERAL DO CEARÁ

- FUNCIONAL:
 - Programas: com mapeamentos
 - Dado **a**, determine o valor de **m(a)** (sempre resultará em uma única resposta)
- LÓGICO:
 - Programas: com relações
 - Dados **a** e **b**, verificar se **R(a, b)** é verdadeiro
 - Dado **a**, encontrar todos os valores para **y** tal que **R(a, y)** é verdadeiro
 - Dado **b**, encontrar todos os valores para **x** tal que **R(x, b)** é verdadeiro
 - Encontre todos os valores para **x** e **y**, tal que **R(x, y)** é verdadeiro

APLICAÇÕES DA PROGRAMAÇÃO LÓGICA



UNIVERSIDADE
FEDERAL DO CEARÁ

- Sistemas Baseados em Conhecimento
 - Sistemas que aplicam mecanismos automatizados de raciocínio para a representação e inferência de conhecimento
- Bancos de Dados “Inteligentes”
 - Sistemas que empregam “agentes” de busca de dados com base em critérios
- Sistemas Especialistas
 - Sistemas que emulam a especialização humana em algum domínio específico.
- Processamento da Linguagem Natural
 - Usada para desenvolvimento de ferramentas para a comunicação homem-máquina em geral e para a construção de interfaces

APLICAÇÕES DA PROGRAMAÇÃO LÓGICA



UNIVERSIDADE
FEDERAL DO CEARÁ

- Indústria de aviação, em soluções de planejamento e escalonamento
 - Coordena 20% do tráfego aéreo do mundo
- Instituto Nacional de Meteorologia (INMET)
 - Previsão do tempo
- Modelagem ambiental
 - Modelos matemáticos para a simulação do desenvolvimento de florestas

APLICAÇÕES DA PROGRAMAÇÃO LÓGICA



UNIVERSIDADE
FEDERAL DO CEARÁ

- Logística
 - Soluções ótimas em tempo real para um fluxo contínuo de ordens de serviço
- Mineração de dados
 - Busca automática em bancos de dados por padrões e relacionamentos significantes

IDEIA BÁSICA DO PROLOG



UNIVERSIDADE
FEDERAL DO CEARÁ

- Descrever a situação de interesse
- Fazer uma pergunta
- Prolog deduz logicamente novos fatos sobre a situação que nós descrevemos
- Prolog retorna suas deduções como respostas

CONSEQUÊNCIAS



UNIVERSIDADE
FEDERAL DO CEARÁ

- Pensar declarativamente, não procedimentalmente
 - Desafiador
 - Requer uma mentalidade diferente
- Linguagem de alto nível
 - Não tão eficiente quanto, digamos, C
 - Bom para prototipagem rápida
 - Útil em muitas aplicações de IA

BASE DE CONHECIMENTO



UNIVERSIDADE
FEDERAL DO CEARÁ

- ?- mulher(maria).
- true
- ?- tocaGuitarra(joana).
- true
- ?- tocaGuitarra(maria).
- False
- ?- tatuada(joana).
- ERROR: predicate tatuada/1 not defined.
- ?- festa.
- true

- mulher(maria).
- mulher(joana).
- mulher(iolanda).
- tocaGuitarra(joana).
- festa.

EXEMPLO



UNIVERSIDADE
FEDERAL DO CEARÁ

- Hierarquia de uma família
- ?- avo(carlos, jose).
- **true.**
- ?- avo(carlos, joao).
- **false.**
- ?- avo(manoel, carlos).
- **false.**
- ?- avo(X, jose).
- X = carlos

```
avo(X, Y) :- pai(X, Z), pai(Z, Y).  
avo(X, Y) :- pai(X, Z), mae(Z, Y).
```

```
pai(carlos, joao).  
pai(joao, jose).
```

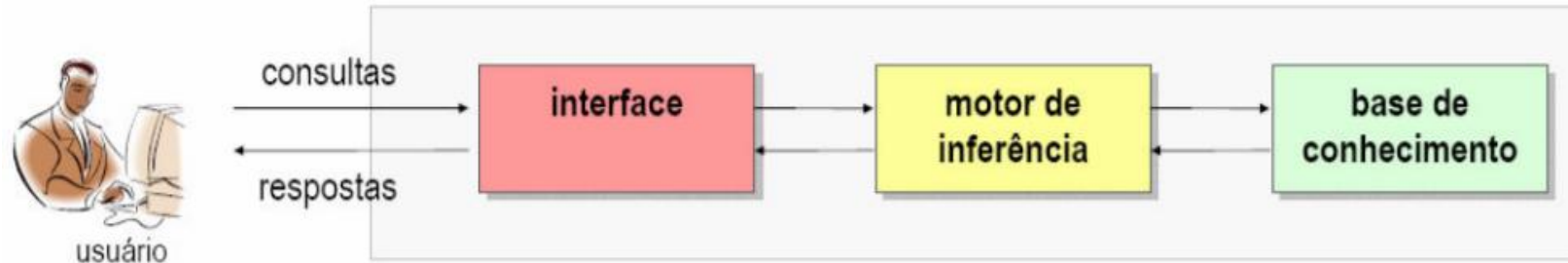
```
mae(maria, joao).
```

INTRODUÇÃO



UNIVERSIDADE
FEDERAL DO CEARÁ

- O prolog é baseado em:
 - Fatos
 - Regras
 - Consultas



PROGRAMA PROLOG



UNIVERSIDADE
FEDERAL DO CEARÁ

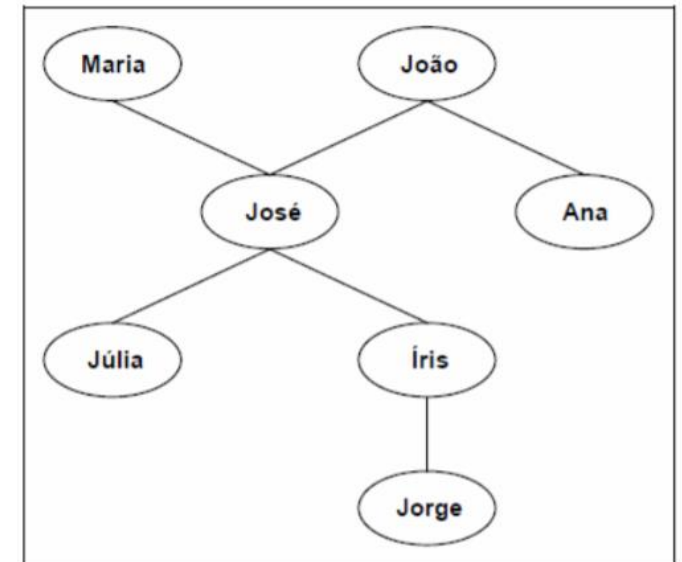
- Fatos
 - São entendidos como relações entre objetos
 - Constituem-se de afirmações que são feitas ao Prolog
 - São verdades nas quais a Prolog irá basear-se para responder as consultas solicitadas
 - Ex: árvore genealógica

PROGRAMA PROLOG



UNIVERSIDADE
FEDERAL DO CEARÁ

- Fatos – cont.
- `progenitor(maria, jose)`





- Regras
 - Especificação de algo que pode ser verdadeiro se algumas condições forem satisfeitas
 - É composta de duas partes: Conclusão (esquerda) e condição (direita)
 - Exemplo: Relação Filho.
 - filho(X,Y) :- progenitor(Y,X).



■ Consultas

- Questionamentos que serão respondidos avaliando-se os fatos e regras
- João é filho de José? $\rightarrow ? \text{filho}(\text{joão}, \text{josé})$.
- Quem é o filho de José? $\rightarrow ?\text{filho}(X, \text{josé})$.
- Quem são os filhos de João $\rightarrow ?\text{filho}(X, \text{joão})$.
 - X é uma variável que representa um objeto desconhecido
 - Variáveis são escritas com a primeira letra em maiúsculo