Introdução à Programação em Prolog

Thiago A. S. Pardo
Solange Rezende

Introdução (1 de 4)

Exercício em duplas

Escrever um **programa completo em C** que armazene quem gosta de quem e que responda sim ou não para perguntas do usuário sobre isso.

- Por exemplo, armazenam-se os fatos de que João gosta de Maria, de que João também gosta de Ana, de que Ana gosta de Pedro, etc. Se o usuário perguntar se João gosta de Ana, a resposta deve ser “sim”, mas a resposta para a pergunta se Ana gosta de João deve ser “não”.

2

Introdução (2 de 4)

Resposta em Prolog

```
gosta(joao,maria).
gosta(joao,ana).
gosta(ana,pedro).
```

3

Introdução (3 de 4)

- Prolog = Programming in Logic
- Linguagem de programação utilizada para resolver problemas envolvendo *objetos*, seus *atributos* e *relações* entre objetos.
- Conceitos básicos: fatos, perguntas, variáveis, conjunções e regras
- Conceitos avançados: listas e recursão

4

Introdução (4 de 4)

- Programação lógica
  - Lógica para definição do que deve ser solucionado
  - Controle para obter as soluções
- Prolog se baseia no cálculo de predicados
  - Sintaxe e semântica bem especificadas e resolução consistente
- A tarefa do programador Prolog é especificar o componente lógico do algoritmo
- O controle da execução é exercido por um sistema de inferência inerente à linguagem lógica

5

Programação em Prolog

Programar em Prolog envolve:

- declarar alguns *fatos* a respeito de objetos e seus relacionamentos
- definir algumas *regras* sobre os objetos e seus relacionamentos e
- fazer *perguntas* sobre os objetos e seus relacionamentos

6

Fatos (1 de 4)

- Expressar em Prolog o fato "João gosta de Maria":
  - objetos: "Maria" e "João"
  - relacionamento: "gosta de", ou simplificando, "gosta"
- Em Prolog:

gosta(joao,maria).

7

Fatos (2 de 4)

- Os nomes de todos os relacionamentos e objetos devem começar com uma letra minúscula
  - Ex: joao, maria, gosta
- O relacionamento é escrito primeiro e os objetos separados por vírgulas e colocados entre parênteses
- O ponto final "." deve seguir o final do fato

Ex: gosta(joao,maria).

8

Fatos (3 de 4)

- A ordem dos objetos é definida arbitrariamente, mas deve ser seguida e usada de forma consistente
  - gosta(joao,maria) significa que "João gosta de Maria"
  - gosta(maria,joao) significa que "Maria gosta de João"
  - gosta(joao,maria) ≠ gosta(maria,joao)

9

Fatos (4 de 4)

- Os nomes dos objetos e relações são arbitrários, ou seja:

  gosta(joao,maria) ou a(b,c)

  são equivalentes desde que "a" signifique "gosta", "b" signifique "joao" e "c" signifique "maria".

- Normalmente, são usados nomes significativos

10

Exercício 1

Expressar em português:

- valioso(ouro).
- femea(jane).
- possui(joao,ouro).
- pai(joao,maria).
- da(joao,livro,maria).

11

Fatos: Terminologia

- Relação: predicado

gosta(joao,maria)

- Objetos: argumentos
- As relações podem ter um número qualquer de argumentos, denominado ***aridade***
- Notação: gosta/2 (fato *gosta* de aridade *2*)

12

Perguntas

Suponha os seguintes fatos (base de dados):

```
gosta(joao,peixe).
gosta(joao,maria).
gosta(maria,livro).
gosta(pedro,livro).
```

Perguntas feitas para Prolog:

```
?- gosta(joao,dinheiro).
no
?- gosta(maria,joao).
no
?- gosta(maria,livro).
yes
?- rei(joao,inglaterra).
no
```

13

Negação por Falha

?- rei(joao,inglaterra).

no

- Como não existe nenhum fato na base sobre a relação “rei”, Prolog responde “no” (ou “false” em alguns prologs), significando que nada foi encontrado que pudesse provar o que foi pedido
- “no” significa “não foi possível provar”

14

Variáveis (1 de 3)

- Variáveis permitem perguntas do tipo: "João gosta do quê?" ou "João gosta de algo?"

```
gosta(joao,peixe).
gosta(joao,maria).
gosta(maria,livro).
gosta(pedro,livro).
```

```
?- gosta(joao,Algo).
Algo = peixe ;
Algo = maria ;
no

?- gosta(maria,X).
X = livro ;
no
```

15

Variáveis (2 de 3)

Uma variável pode estar

- instanciada
  - quando a variável já referencia algum objeto
- não-instanciada
  - quando a variável não referencia nenhum objeto, ou seja, quando o objeto a que ela referencia ainda não é conhecido

- Variáveis começam com letra maiúscula ou com o símbolo de sublinhado "_"

16

Variáveis (3 de 3)

- Quando uma variável é usada numa pergunta, Prolog procura por todos os fatos tentando encontrar um objeto no qual a variável possa ser instanciada
- Quando uma solução é encontrada, ela é mostrada
- Se o usuário estiver satisfeito com a resposta, basta digitar "." ou *enter*
- Se desejar mais respostas, usa-se ponto-e-vírgula ";"

17

Exercício 2

- Base de fatos:

```
gosta(joao,peixe).
gosta(joao,maria).
gosta(maria,livro).
gosta(pedro,livro).
gosta(maria,flor).
gosta(maria,vinho).
```

Qual o resultado das seguintes perguntas?

```
?- gosta(maria,X).
?- gosta(X,livro).
?- gosta(Quem,Oque).
?- gosta(X,Y).
?- gosta(X,X).
?- gosta(_a,_b).
?- gosta(A,peixe).
```

18

Conjunções (1 de 2)

- Permitem expressar relacionamentos mais complexos
- "João gosta de Maria" e "Maria gosta de João" pode ser expresso pela pergunta:

```
?- gosta(joao,maria), gosta(maria,joao).
```

- Ou seja, a conjunção é indicada usando-se vírgula entre as relações (cláusulas)

19

Conjunções (2 de 2)

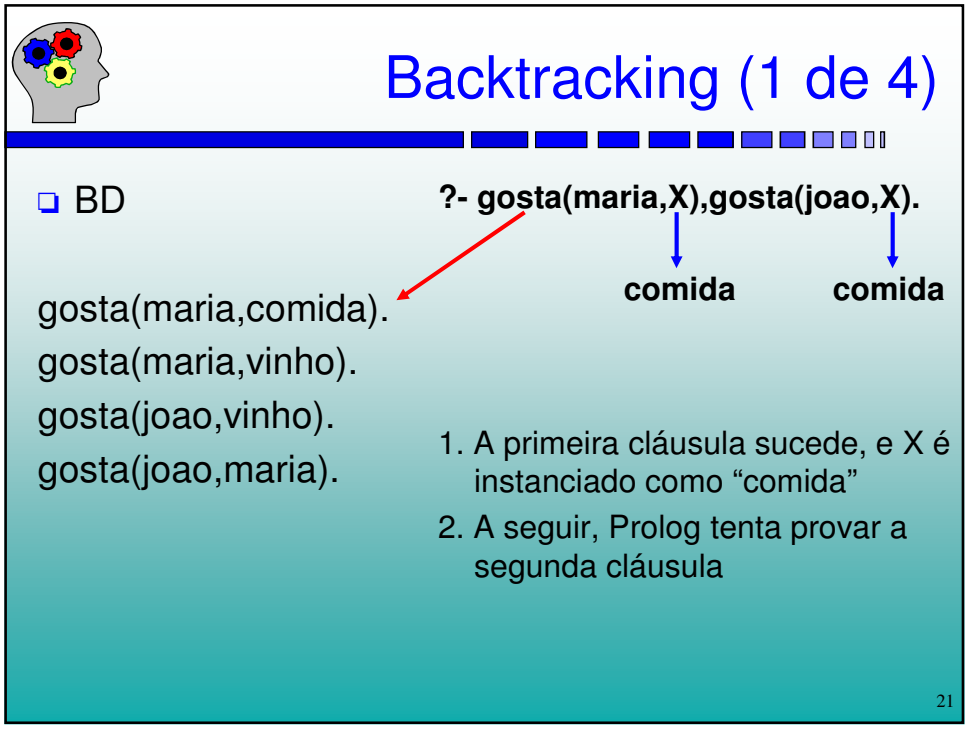
Ex: "Existe algo que João e Maria gostam?"

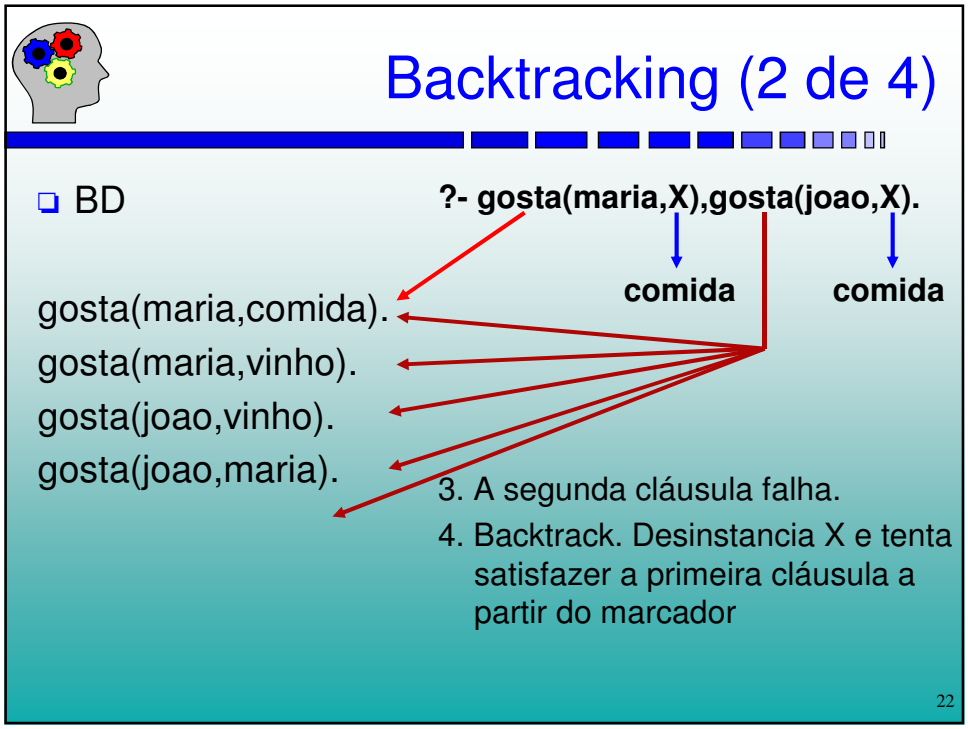
1. Encontre um X tal que João goste
2. Então, verifique se Maria também gosta de X

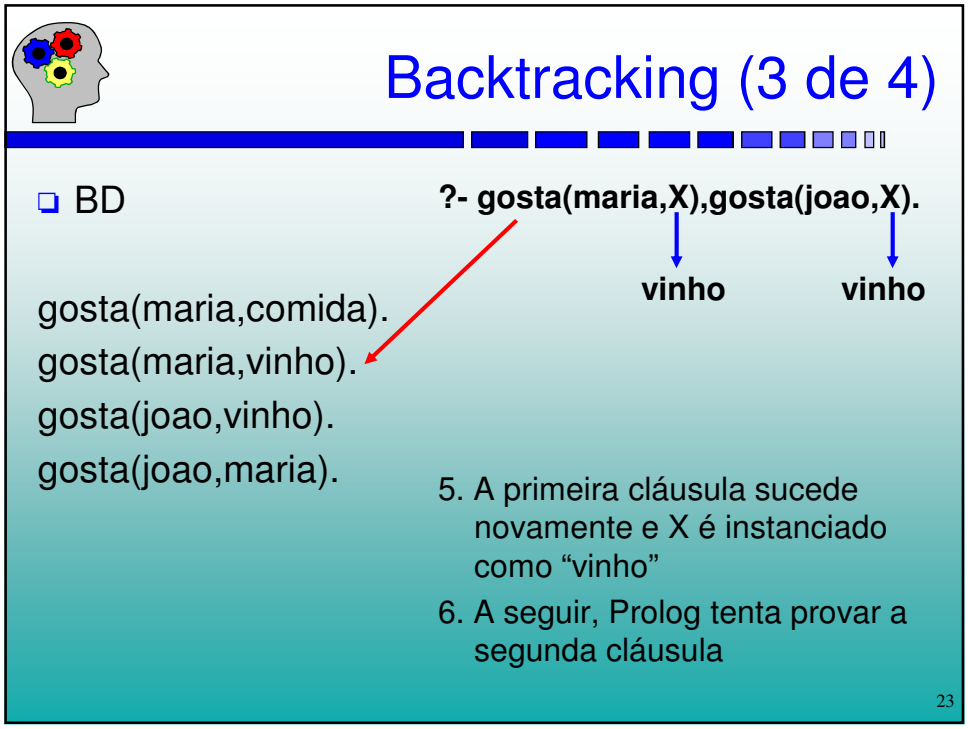
```
?- gosta(joao,X), gosta(maria,X).
```

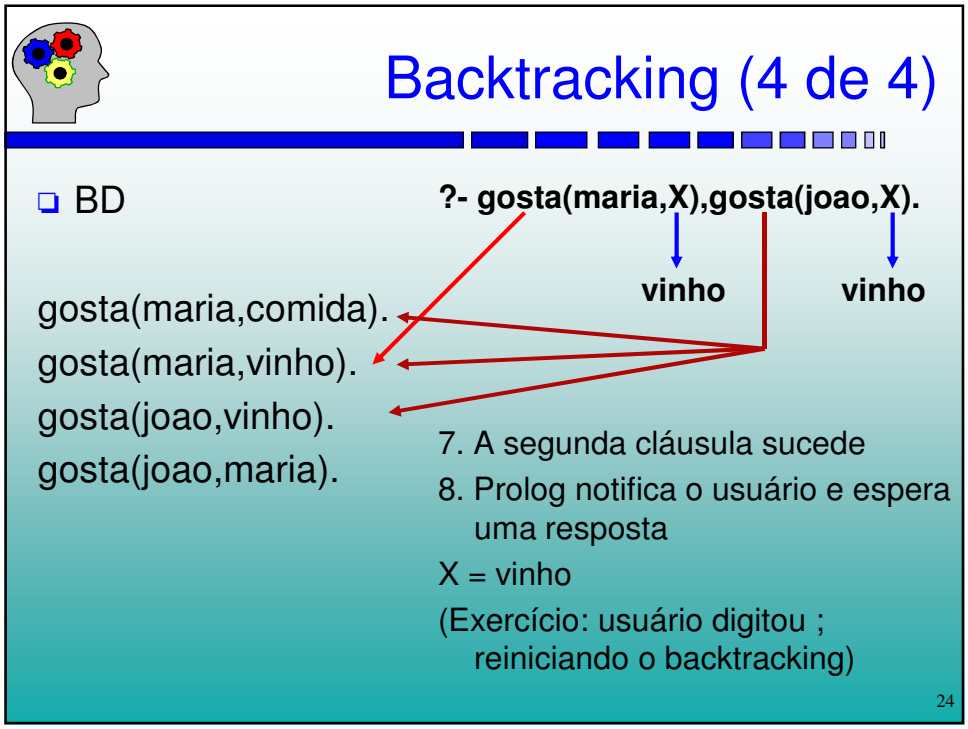
- Prolog tenta satisfazer a primeira cláusula. Se a primeira cláusula está na base, então Prolog marca o local na base e tenta satisfazer a segunda cláusula. Se a segunda cláusula é satisfeita, então Prolog marca o local na base e exibe a solução que satisfaz ambas as cláusulas.
- Cada cláusula tem seu próprio marcador na base de dados

20


Backtracking (1 de 4)
BD
?- gosta(maria,X),gosta(joao,X).
comida
comida
gosta(maria,comida).
gosta(maria,vinho).
gosta(joao,vinho).
gosta(joao,maria).
1. A primeira cláusula sucede, e X é instanciado como "comida"
2. A seguir, Prolog tenta provar a segunda cláusula
21


Backtracking (2 de 4)
BD
?- gosta(maria,X),gosta(joao,X).
comida
comida
gosta(maria,comida).
gosta(maria,vinho).
gosta(joao,vinho).
gosta(joao,maria).
3. A segunda cláusula falha.
4. Backtrack. Desinstancia X e tenta satisfazer a primeira cláusula a partir do marcador
22


Backtracking (3 de 4)
BD
?- gosta(maria,X),gosta(joao,X).
vinho
vinho
gosta(maria,comida).
gosta(maria,vinho).
gosta(joao,vinho).
gosta(joao,maria).
5. A primeira cláusula sucede novamente e X é instanciado como “vinho”
6. A seguir, Prolog tenta provar a segunda cláusula
23


Backtracking (4 de 4)
BD
?- gosta(maria,X),gosta(joao,X).
vinho
vinho
gosta(maria,comida).
gosta(maria,vinho).
gosta(joao,vinho).
gosta(joao,maria).
7. A segunda cláusula sucede
8. Prolog notifica o usuário e espera uma resposta
X = vinho
(Exercício: usuário digitou ; reiniciando o backtracking)
24

Regras (1 de 2)

- Como expressar o fato que João gosta de todas as pessoas?

gosta(joao,maria).
gosta(joao,pedro).
gosta(joao,david).
gosta(joao,jaqueline).
gosta(joao,ana).
...

- Esta forma de expressar torna-se tediosa para centenas de pessoas
- Solução: escrever uma regra que diz *João gosta de todas as pessoas*
- A regra é bem mais compacta que o conjunto de fatos

25

Regras (2 de 2)

- Em Prolog, regras são usadas para especificar que um fato *depende* de um grupo de outros fatos
- Em português, usa-se a palavra "se" para expressar uma regra

Exemplos:

- X é irmã de Y se:
  X é mulher e
  X e Y possuem os mesmos pais.
- X é um pássaro se:
  X é um animal e
  X tem penas.

26

De Português para Prolog

- João gosta de qualquer pessoa que gosta de vinho
  - João gosta de X se X gosta de vinho.
    - gosta(joao,X) :-
      gosta(X,vinho).

27

Regra: Cabeça e Cauda

- Em Prolog, uma regra consiste de uma cabeça e uma cauda (ou corpo)
- A cabeça e a cauda são conectadas pelo símbolo :-, denominado *neck*, formado por dois pontos e hífen
- :- é pronunciado "se"

gosta(joao,X) :- gosta(X,vinho).

equivale a

João gosta de X se X gosta de vinho.

28

Exercício 3

- Identifique a cabeça e cauda de cada regra
- Expressar cada regra em Português

```
gosta(joao,X) :-
        gosta(X,vinho),
        gosta(X,comida).
gosta(joao,X) :-
        mulher(X),
        gosta(X,vinho).
```

29

De Português para Prolog

- João gosta de qualquer pessoa que não goste de violência
  - João gosta de X se X não gosta de violência
    - gosta(joao,X) :-
      \+ gosta(X,violencia).
  - Outras formas de se fazer?

30

Exercício 4

- Usando a base ao lado, defina a regra: uma pessoa pode roubar um objeto se essa pessoa é um ladrão e ela gosta de um objeto
- Formule a pergunta "João rouba o quê?"

```
ladrao(joao).
ladrao(pedro).
gosta(maria,comida).
gosta(maria,vinho).
gosta(joao,rubi).
gosta(joao,X) :-
   gosta(X,vinho).
```

31

Exercício 5

- Implemente um programa em prolog (base de conhecimento e regras) para determinar se uma pessoa é irmã de outra, sabendo o seguinte:
  - uma pessoa X é irmã de uma pessoa Y se X é mulher e X e Y possuem os mesmos pais

32

Exercício 6

- **Estenda** o programa anterior para considerar o caso de irmãos que não sabem que têm os mesmos pais, pois foram separados no início de suas vidas

33

Exercício 7

- Monte uma base de dados sobre as mulheres de uma família e suas relações de descendências
  - Deve-se poder consultar se 1 pessoa descende direta ou indiretamente de outra

34

Fim