

# **APRENDIZAGEM E EXTRAÇÃO DE CONHECIMENTO**

2017/2018

# PROGRAMAÇÃO EM LÓGICA DE HORN

## Tipos de negação

- negação forte ( $\neg$ )
- negação fraca ou por falha na prova (*não*)
- falsidade (?)

Enunciação do predicado *filho*:

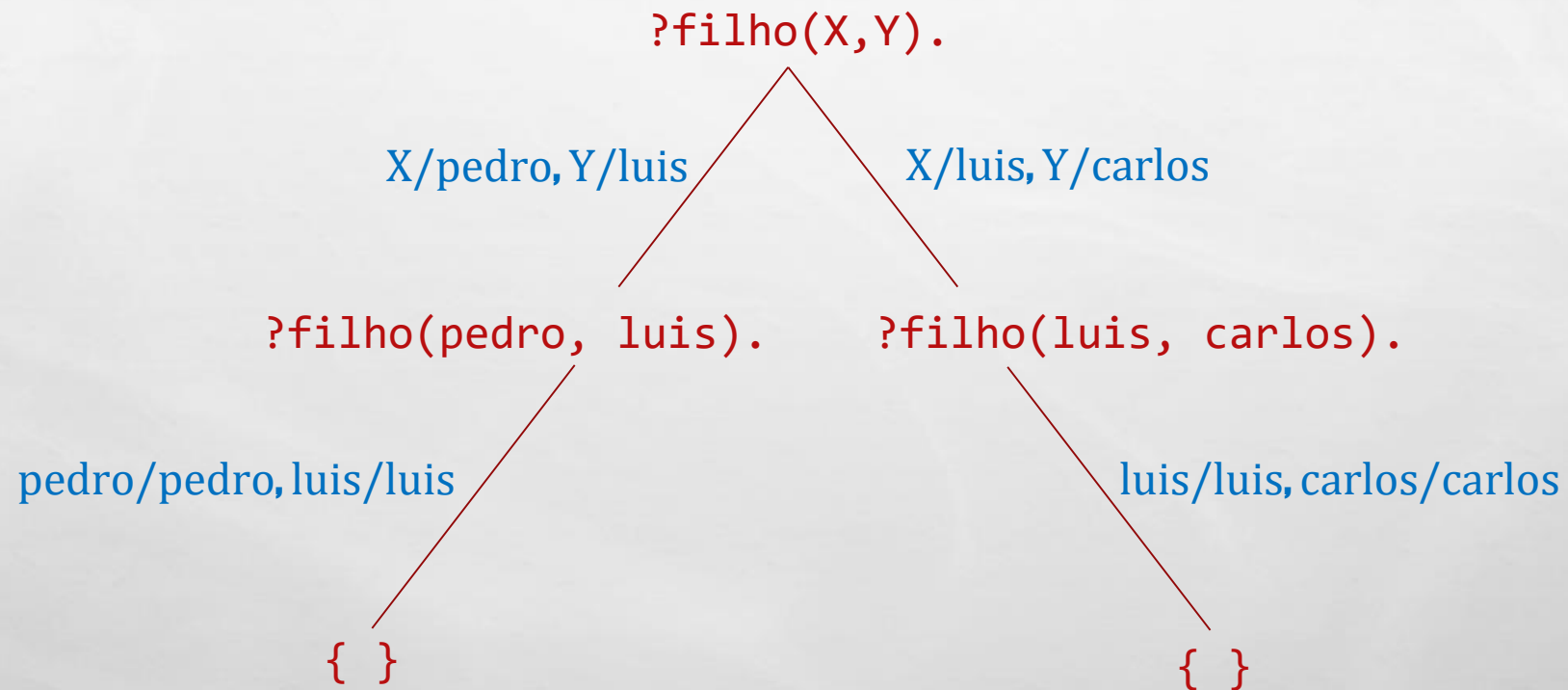
*filho*:  $X, Y \rightarrow \{V, F\}$

?*filho*(X,Y).  $\Leftrightarrow$  Quem é filho de quem?

# PROGRAMAÇÃO EM LÓGICA DE HORN

$$\left[ \begin{array}{l} \neg \text{filho}(x,y): - \text{não filho}(x,y). \\ \text{filho}(\text{pedro}, \text{luis}). \\ \text{filho}(\text{luis}, \text{carlos}). \\ \neg \text{filho}(\text{manuel}, \text{luis}). \end{array} \right] \cup \left[ \text{?filho}(X,Y). \right] = \left[ \right]$$

# ÁRVORE DE PROVA



# RELAÇÃO DE DERIVABILIDADE $\vdash$

$$\vdash = \left\{ \langle P_{\text{programa}}, t_{\text{termo}} \rangle \mid P \subseteq \text{Linguagem Programação Lógica}, t \in \text{LPL}, e \right\}$$

$t$  é derivado de  $P$  através da utilização das regras de derivação

## Regra de Derivação *modus tollens*

$$\frac{? p. \quad p \text{ se } q.}{? q.}, p \text{ é o termo}$$

# PROGRAMAÇÃO EM LÓGICA DE HORN

Conjuntos são aproximados por **listas**:

- $\{ \} \cong [ ]$
- $\{1\} \cong [1|[ ]]$
- $\{1, 2, 3\} \cong [1, 2, 3] \cong [1, 2|[3]] \cong [1|[2, 3]]$

# PREDICADO soluções

## Argumentos

1. arquétipo da solução (que forma tem a solução)
2. teorema(s) a demonstrar
3. obter a solução por forma a manipulá-la

soluções:  $X_{\text{arquétipo da solução}}, T_{\text{teorema}}, S_{\text{solução}} \rightarrow \{V, F\}$

# PROGRAMAÇÃO EM LÓGICA DE HORN

filho:  $X, Y \rightarrow \{V, F\}$

$$\left\{ \begin{array}{l} \neg \text{filho}(X, Y): - \text{não filho}(X, Y), \text{ não exceção filho}(X, Y). \\ \text{filho}(\text{joao}, \text{carlos}). \\ \text{filho}(\text{carlos}, \text{pedro}). \end{array} \right\} \cup \left\{ ?\text{filho}(X, Y). \right\} = \left\{ \right\}$$

( $\forall \text{termo},$   
 $\text{termo} = \neg \text{termo}$ )



# ALGORITMO soluções

início

enquanto  $P \cup \{?T\} = \{ \}$  fazer

$?T$ ; ( $\Leftrightarrow$  demonstrar  $T \cong ?filho(X,Y).$ )

$?inserir\ temp(arquétipo\ já\ instanciado).$  ( $\Leftrightarrow$  e.g.,  $temp([joao, carlos]).$ )

enquanto  $P \cup \{?obter([ ], Z)\} = \{ \}$  fazer

$?remove(temp(Y));$

$?obter([Y|W],Z).$

$?obter(Y,Y).$

fim

$S = [[carlos, pedro], [joao, carlos]].$

# ALGORITMO soluções

```
¬soluções(X,T,S) ← não soluções(X,T,S), não exceção soluções(X,T,S).  
soluções(X,T,S) ← T, assert(temp(X)), fail.  
soluções(X,T,S) ← obter([ ],S).
```

```
¬obter(X,Y) ← não obter(X,Y), não exceção obter(X,Y).  
obter(X,Y) ← retract(temp(Y)), obter([Y|W],Z).  
obter(Y,Y).
```

```
¬temp(X): – não temp(X), não exceção temp(X).  
temp([joao, carlos]).  
temp([carlos, pedro]).
```