



# **Paradigma de Linguagem de Programação**

Programação Lógica

Prolog



# Prolog

- Criada por Robert Kowalski, Alain Colmerauer e Phillippe Roussel, na década de 70.
- Algumas aplicações :
  - ✓ Sistemas Baseados em Conhecimento.
  - ✓ Sistemas Especialistas.
  - ✓ Processamento de Linguagem Natural.
- Não é uma linguagem procedural como C++ ou Pascal, e sim uma linguagem declarativa (não é preciso definir passo a passo o que deve ser feito e sim descrever o problema).
- Pode provar teses a partir de hipóteses.
- Inclui predicados, conectivos lógicos e regras de inferência (princípio da resolução).

**Princípio da resolução:** regra de inferência que dá origem a uma técnica de demonstração por dedução de uma contradição (refutação).





# Conceitos básicos sobre Prolog

- Na lógica de predicados usamos regras de inferência para demonstrar que uma tese é consequência de certas hipóteses.
- As declarações de um programa formam uma base de dados e para determinar se uma tese (consulta de um usuário à base de dados) é verdadeira ou não, Prolog aplica suas regras de inferência nesta base.
- Um programa em Prolog (***Programming Logic***) é uma declaração de hipóteses e normalmente se divide em:
  - ✓ Declaração de fatos sobre objetos e suas relações.
  - ✓ Definição de regras.
  - ✓ Consultas.

# Prolog

- As bases de dados convencionais descrevem apenas os fatos, ao passo que Prolog registra também as regras. E baseando-se nestas regras pode-se gerar novos fatos.

Fato	Regra	Dedução de um novo fato
Rufos é um vira-lata.	Todo vira-lata é um cachorro.	Rufos é um cachorro.

- Operadores:

Linguagem natural	Cálculo de predicados	Prolog
E	$\wedge$	,
OU	$\vee$	;
SE	$\leftarrow$	$::-$
NÃO	$\neg$	<b>not</b>

# Prolog

- As sentenças são expressas por cláusulas e as cláusulas se dividem em fatos e regras.
  - ✓ **Fato** é a declaração de uma verdade incondicional (os fatos permitem definir os predicados).
  - ✓ **Regra** é a condição que deve ser satisfeita para que uma declaração seja considerada verdadeira.
- Exemplos:

rato(jerry).

Este é um exemplo de FATO:

"rato" é o predicado (cabeça) e "jerry" é o argumento do predicado .

roedor(X) :- rato(X); toupeira(X).

Este é um exemplo de REGRA:

Observe o uso do operador "OU" (;) para determinar a validade da regra.

# Exemplo 1

```
/* 1) Fatos */
progenitor(maria, rodrigo).      /* Predicado binário */
progenitor(emmanuel, rodrigo).

progenitor(rodrigo, julia).
progenitor(silvia, julia).

progenitor(rodrigo, jorge).
progenitor(ana, jorge).

masculino(emmanuel). /* Predicado unário */
masculino(rodrigo).
masculino(jorge).

feminino(maria).
feminino(ana).
feminino(julia).
feminino(silvia).

/* 2) Regras */
irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
irma(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).

pai(X, Y) :- progenitor(X, Y), masculino(X).
mae(X, Y) :- progenitor(X, Y), feminino(X).

/* 3) Consultas */
?- mae(ana, jorge).
```

The screenshot shows the Strawberry Prolog (Light edition) interface. The top window is titled "Primeiro.pro" and contains the Prolog code. A "Run" button is visible in the toolbar above the code area. Below it, an "Output" window displays the results of the compilation and execution. The output shows:

```
/* 3) Consultas */

?- mae(ana, jorge).

Yes.

Compiling the file:
C:\Elson\2009 - 02\Disciplinas\Paradigmas de
0 errors, 0 warnings.

Yes.
```

Observe que ao executarmos o código, o interpretador do Strawberry Prolog nos diz "Yes", ou seja, "Ana é mãe de Jorge". Procure analisar os fatos e regras, assim começará a perceber a recursividade (backtracking) do mecanismo de inferência.

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH web interface. At the top, there is a navigation bar with icons for gear, user, and search, followed by "250 users online". Below the navigation bar is a search bar with a magnifying glass icon and a dropdown menu. The main area has a large, stylized owl logo. On the left, there is a sidebar with a "New tab" button, a "Create a" dropdown set to "Program", and a "here" button. Below this, there is a "based on" dropdown with options: Empty, Student, CLP, s(CASP), and profile. A search input field contains "user:'me'" with a clear button, filter dropdown, type dropdown, and search button. Below the search input, a red warning icon says "No matching files" and provides instructions for new users. A link "help on search" is also present. At the bottom, there is a footer with links for Examples, History, Solutions, and a "Run" button.

SWISH

File ▾ Edit ▾ Examples ▾ Help ▾

250 users online

Search

New tab +

Create a Program Notebook here

based on Empty Student CLP s(CASP) profile

user:'me' X Filter ▾ Type ▾ 🔍

⚠ No matching files

If you are a new user you may

- Use the Examples menu from the navigation bar
- Use the Program or Notebook button above

help on search

?-

Examples ▾ History ▾ Solutions ▾  table results Run

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

A screenshot of the SWISH web interface. At the top, there's a navigation bar with 'File', 'Edit', 'Examples', and 'Help'. Below the navigation bar, a yellow tooltip box with a black border and rounded corners contains the text 'Clique em "Program".' A large red arrow points from the bottom left towards the 'Program' button. The main area has a light gray background featuring a large, stylized owl logo. On the left, there's a search bar with the placeholder 'user:"me"' and a 'New tab' button. Below the search bar, a message says 'No matching files' with a warning icon. It also provides instructions for new users: 'If you are a new user you may' followed by two bullet points: 'Use the Examples menu from the navigation bar' and 'Use the Program or Notebook button above'. There's a link 'help on search' at the bottom of this section. At the very bottom, there are links for 'Examples', 'History', 'Solutions', and a 'Run' button.

Clique em "Program".

SWISH

File ▾ Edit ▾ Examples ▾ Help ▾

New tab +

Create a **Program** Notebook here

based on Empty Student CLP s(CASP) profile

user:"me" X Filter ▾ Type ▾ Q

**No matching files**

If you are a new user you may

- Use the Examples menu from the navigation bar
- Use the Program or Notebook button above

help on search

?-

Examples ▾ History ▾ Solutions ▾  table results **Run**

# SWISH - Prolog

- <https://swish.swi-prolog.org>

The screenshot shows the SWISH Prolog interface. At the top, there's a navigation bar with 'File', 'Edit', 'Examples', and 'Help' menus, and a search bar. A yellow callout box says 'Cole os fatos aqui.' (Paste facts here). Below the code area, there's a status bar showing '257 users online'. On the right, there's a large owl logo.

The main area contains the following Prolog code:

```
1  /* 1) Fatos */
2  progenitor(maria, rodrigo).      /* Predicado binário */
3  progenitor(emmanuel, rodrigo).
4
5  progenitor(rodrigo, julia).
6  progenitor(silvia, julia).
7
8  progenitor(rodrigo, jorge).
9  progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irma(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(X, Y) :- progenitor(X, Y), feminino(X).
```

Two red arrows point from the text 'Cole os fatos aqui.' and 'Cole as regras aqui.' to the respective sections of the code. The bottom right corner has buttons for 'table results' and 'Run!'

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, there is a code editor window titled "Program" containing Prolog code. The code includes facts about progenitors and gender, and rules for sibling and parent relationships. On the right, there is a query window with a yellow callout bubble and a red arrow pointing to it.

/\* 1) Fatos \*/  
progenitor(maria, rodrigo). /\* Predicado binário \*/  
progenitor(emmanuel, rodrigo).  
  
progenitor(rodrigo, julia).  
progenitor(silvia, julia).  
  
progenitor(rodrigo, jorge).  
progenitor(ana, jorge).  
  
masculino(emmanuel). /\* Predicado unário \*/  
masculino(rodrigo).  
masculino(jorge).  
  
feminino(maria).  
feminino(ana).  
feminino(julia).  
feminino(silvia).  
  
/\* 2) Regras \*/  
irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).  
irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).  
  
pai(X, Y) :- progenitor(X, Y), masculino(X).  
mae(X, Y) :- progenitor(X, Y), feminino(X).

Cole as consultas aqui.  
Não esqueça de remover o "?-".

?- /\* 3) Consultas \*/  
mae(ana, jorge).

Examples▲ History▲ Solutions▲

□ table results Run!

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH web-based Prolog environment. At the top, there's a navigation bar with links for File, Edit, Examples, and Help. To the right of the examples link is a search bar and a user count of "255 users online". Below the navigation bar is a toolbar with icons for file operations and a bell通知 icon.

The main area is divided into two panes. The left pane is a code editor titled "Program" containing the following Prolog code:

```
1 /* 1) Fatos */
2 progenitor(maria, rodrigo).      /* Predicado binário */
3 progenitor(emmanuel, rodrigo).
4
5 progenitor(rodrigo, julia).
6 progenitor(silvia, julia).
7
8 progenitor(rodrigo, jorge).
9 progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irma(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(X, Y) :- progenitor(X, Y), feminino(X).
```

The right pane contains a large owl logo and a yellow speech bubble with the text "Execute, clicando em "Run!".". Below the owl is a query window with the following text:

```
?- /* 3) Consultas */
mae(ana, jorge).
```

At the bottom right of the interface, there are buttons for "table results" and a prominent red "Run" button.

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, there is a code editor window titled "Program" containing Prolog code. The code includes facts about progenitors and gender, and rules for sibling and parent relationships. On the right, the execution environment shows the query `?- /* 3) Consultas */ mae(ana, jorge).` and its result `true`. A red arrow points to the result line. A yellow callout bubble with black text says: "Resultado da consulta apresentado na saída." (Result of the query displayed in the output). The top of the interface has a navigation bar with File, Edit, Examples, Help, and a search bar. There are also icons for user status (258 users online), search, and other system functions.

```
1  /* 1) Fatos */
2  progenitor(maria, rodrigo).      /* Predicado binário */
3  progenitor(emmanuel, rodrigo).
4
5  progenitor(rodrigo, julia).
6  progenitor(silvia, julia).
7
8  progenitor(rodrigo, jorge).
9  progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(X, Y) :- progenitor(X, Y), feminino(X).
```

Resultado da consulta apresentado na saída.

```
?- /* 3) Consultas */
mae(ana, jorge).
```

true

Examples History Solutions

Run!

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, there is a code editor window titled "Program" containing Prolog code. A yellow speech bubble points to the code with the text "Alteração da consulta.". On the right, there is a query window and a results window. The query window contains a query and its result. A red arrow points from the "Alteração da consulta." bubble to the query window. A yellow speech bubble points to the results window with the text "Resultado da alteração consulta.". The results window shows the query and its true result. At the bottom, there are navigation buttons for "Examples", "History", "Solutions", and a "Run" button.

Alteração da consulta.

```
1 /* 1) Fatos */
2 progenitor(maria, rodrigo).      /* Predicado binário */
3 progenitor(emmanuel, rodrigo).
4
5 progenitor(rodrigo, julia).
6 progenitor(silvia, julia).
7
8 progenitor(rodrigo, jorge).
9 progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(Y, X) :- progenitor(X, Y), feminino(X).
```

Resultado da alteração consulta.

\*/ 3) Consultas \*/ write('Ana é mãe de Jorge?'), mae(ana, jorge).

Ana é mãe de Jorge?  
true

?- /\* 3) Consultas \*/  
write('Ana é mãe de Jorge? '),
mae(ana, jorge).

Examples▲ History▲ Solutions▲

Run!

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, there is a code editor window titled "Program" containing Prolog code. A yellow callout bubble points to the code with the text "Alteração da consulta." (Change in query). On the right, there is a results window showing a query and its answer. The top of the interface has a navigation bar with "File", "Edit", "Examples", "Help", a search bar, and user statistics ("378 users online"). A large cartoon owl is visible in the background of the interface.

```
1 /* 1) Fatos */
2 progenitor(maria, rodrigo).      /* Predicado binário */
3 progenitor(emmanuel, rodrigo).
4
5 progenitor(rodrigo, julia).
6 progenitor(silvia, julia).
7
8 progenitor(rodrigo, jorge).
9 progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(X, Y) :- progenitor(X, Y), feminino(X).
```

Alteração da consulta.

```
?- /* 3) Consultas */
X = ana,
mae(X, Y).
```

Examples ▾ History ▾ Solutions ▾

□ table results Run!

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, there is a code editor window titled "Program" containing Prolog code. A yellow speech bubble with the text "Resultado da alteração consulta." points to the query results window on the right. A large red arrow also points from the speech bubble towards the results window.

**Program**

```
1 /* 1) Fatos */
2 progenitor(maria, rodrigo).      /* Predicado binário */
3 progenitor(emmanuel, rodrigo).
4
5 progenitor(rodrigo, julia).
6 progenitor(silvia, julia).
7
8 progenitor(rodrigo, jorge).
9 progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irma(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(X, Y) :- progenitor(X, Y), feminino(X).
```

**Consulta**

```
/* 3) Consultas */ X = ana, mae(X, Y).
```

X = ana,  
Y = jorge

?- /\* 3) Consultas \*/  
X = ana,  
mae(X, Y).

Examples History Solutions

Resultados

375 users online

Search

Run!

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, there is a code editor window titled "Program" containing Prolog code. A yellow callout bubble points from the code area to the right pane, which displays the results of a query. A large red arrow points from the callout bubble towards the right pane.

**Code Editor (Left):**

```
1 /* 1) Fatos */
2 progenitor(maria, rodrigo).      /* Predicado binário */
3 progenitor(emmanuel, rodrigo).
4
5 progenitor(rodrigo, julia).
6 progenitor(silvia, julia).
7
8 progenitor(rodrigo, jorge).
9 progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(X, Y) :- progenitor(X, Y), feminino(X).
```

**Right Pane (Results):**

```
?- /* 3) Consultas */
progenitor(X, jorge).
```

**Annotations:**

- A yellow callout bubble with the text "Alteração da consulta." is positioned above the red arrow, pointing towards the right pane.
- A large red arrow points from the callout bubble towards the right pane, indicating the flow of the query or result.

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, there is a code editor window displaying a Prolog program. The code includes facts about progenitors and gender, and rules for sibling and parent relationships. On the right, the query results are displayed in a results window. A red arrow points from a yellow speech bubble labeled "Resultado da alteração consulta." to the results window. The results window shows the query `?- /* 3) Consultas */ progenitor(X, jorge).` and its answer `X = rodrigo`. Below the results window are buttons for "Next", "10", "100", "1,000", and "Stop". At the bottom of the interface, there are links for "Examples", "History", "Solutions", and buttons for "table results" and "Run".

```
/* 1) Fatos */
progenitor(maria, rodrigo).      /* Predicado binário */
progenitor(emmanuel, rodrigo).

progenitor(rodrigo, julia).
progenitor(silvia, julia).

progenitor(rodrigo, jorge).
progenitor(ana, jorge).

masculino(emmanuel). /* Predicado unário */
masculino(rodrigo).
masculino(jorge).

feminino(maria).
feminino(ana).
feminino(julia).
feminino(silvia).

/* 2) Regras */
irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
irma(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).

pai(X, Y) :- progenitor(X, Y), masculino(X).
mae(X, Y) :- progenitor(X, Y), feminino(X).
```

Resultado da alteração consulta.

```
?- /* 3) Consultas */ progenitor(X, jorge).
```

X = rodrigo

Next 10 100 1,000 Stop

```
?- /* 3) Consultas */
progenitor(X, jorge).
```

Examples History Solutions

table results Run

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, the 'Program' tab is selected, displaying the following Prolog code:

```
1  /* 1) Fatos */
2  progenitor(maria, rodrigo).      /* Predicado binário */
3  progenitor(emmanuel, rodrigo).
4
5  progenitor(rodrigo, julia).
6  progenitor(silvia, julia).
7
8  progenitor(rodrigo, jorge).
9  progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irma(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(X, Y) :- progenitor(X, Y), feminino(X).
```

A yellow callout bubble with the text "Clique em 'Next'." points to the 'Next' button in the query evaluator window on the right. A large red arrow also points to this button. The query evaluator window shows the following:

/\* 3) Consultas \*/ progenitor(X, jorge).

X = rodrigo

Next 10 100 1,000 Stop

?- /\* 3) Consultas \*/
progenitor(X, jorge).

At the bottom of the interface, there are links for 'Examples', 'History', 'Solutions', and buttons for 'table results' and 'Run'.

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH Prolog interface. On the left, there is a code editor window titled "Program" containing Prolog code. A yellow speech bubble points to the right side of the interface with the text "Próximo resultado da consulta." (Next result of the query). On the right, a results window titled "/\* 3) Consultas \*/ progenitor(X, jorge)." displays two results: "X = rodrigo" and "X = ana". A red arrow points from the speech bubble to the "X = rodrigo" result. The interface includes a navigation bar with "File", "Edit", "Examples", and "Help" menus, a search bar, and various icons.

```
1 /* 1) Fatos */
2 progenitor(maria, rodrigo).      /* Predicado binário */
3 progenitor(emmanuel, rodrigo).
4
5 progenitor(rodrigo, julia).
6 progenitor(silvia, julia).
7
8 progenitor(rodrigo, jorge).
9 progenitor(ana, jorge).
10
11 masculino(emmanuel). /* Predicado unário */
12 masculino(rodrigo).
13 masculino(jorge).
14
15 feminino(maria).
16 feminino(ana).
17 feminino(julia).
18 feminino(silvia).
19
20
21 /* 2) Regras */
22 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
23 irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).
24
25 pai(X, Y) :- progenitor(X, Y), masculino(X).
26 mae(X, Y) :- progenitor(X, Y), feminino(X).
```

Próximo resultado da consulta.

/\* 3) Consultas \*/ progenitor(X, jorge).

X = rodrigo

X = ana

?- /\* 3) Consultas \*/ progenitor(X, jorge).

Examples▲ History▲ Solutions▲

table results Run!



# SWISH - Prolog

- <https://swish.swi-prolog.org/>

```
/* 1) Fatos */
```

```
/* 2) Regras */
```

```
% If-Else:
```

```
maior(X, Y) :- X >= Y, write('-> maior ou igual').  
maior(X, Y) :- X < Y, write('-> menor').
```

```
% If-Elif-Else:
```

```
compara(X, Y) :- X > Y, write('-> maior').  
compara(X, Y) :- X =:= Y, write('-> igual').  
compara(X, Y) :- X < Y, write('-> menor').
```

```
/* 3) Consultas */
```

```
write('5 > 3? '), maior(5, 3), nl,  
write('5 < 7? '), maior(5, 7), nl,  
write('10 > 9? '), compara(10, 9), nl,  
write('9 == 9? '), compara(9, 9), nl,  
write('8 < 9? '), compara(8, 9).
```

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH web-based Prolog environment. The top navigation bar includes links for File, Edit, Examples, Help, and a search bar. A status bar indicates "388 users online". The main area has tabs for Program (+) and Examples. The current tab displays the following Prolog code:

```
1 /* 1) Fatos */
2
3 /* 2) Regras */
4
5 % If-Else:
6 maior(X, Y) :- X >= Y, write('-> maior ou igual').
7 menor(X, Y) :- X < Y, write('-> menor').
8
9 % If-Elif-Else:
10 compara(X, Y) :- X > Y, write('-> maior').
11 compara(X, Y) :- X == Y, write('-> igual').
12 compara(X, Y) :- X < Y, write('-> menor').
```

The right side of the interface shows the results of a query:

```
?- /* 3) Consultas */
write('5 > 3? '), maior(5, 3), nl, write('5 < 7? '), maior(5, 7), nl,
write('10 > 9? '), compara(10, 9), nl, write('9 == 9? '), compara(9, 9), nl,
write('8 < 9? '), compara(8, 9).
```

Output:

```
5 > 3? -> maior ou igual
5 < 7? -> menor
10 > 9? -> maior
9 == 9? -> igual
8 < 9? -> menor
true
```

At the bottom, there are buttons for Next, Stop, and a dropdown menu set to 1000. The footer includes links for Examples, History, Solutions, and Run.

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH web-based Prolog environment. On the left, the code editor displays the following Prolog code:

```
1 /* 1) Fatos */
2
3
4 /* 2) Regras */
5 nao_vota(X) :-
6   X < 16;
7   X > 70.
```

Three yellow callout boxes with black outlines point from the right side of the screen towards the results pane. The top callout box contains the text "Resultado para 15 anos." A large red arrow points from this box to the results pane. The middle callout box contains "Resultado para 25 anos." and the bottom one contains "Resultado para 75 anos." Both of these boxes also have red arrows pointing to the results pane.

The results pane on the right shows the following interactions:

- /\* 3) Consultas \*/ write('15 anos'), nao\_vota(15), nl.  
15 anos  
true  
Next 10 100 1,000 Stop
- /\* 3) Consultas \*/ write('25 anos'), nao\_vota(25), nl.  
25 anosfalse
- /\* 3) Consultas \*/ write('75 anos'), nao\_vota(75), nl.  
75 anos  
true

At the bottom of the results pane, there is a query:  
?- /\* 3) Consultas \*/  
write('75 anos'), nao\_vota(75), nl.

At the very bottom of the interface, there are navigation buttons: Examples▲ History▲ Solutions▲, a checkbox for "table results", and a "Run!" button.

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH interface with the following components:

- Header:** SWISH logo, File, Edit, Examples, Help, 348 users online, Search, and various icons.
- Code Editor:** A tab labeled "Program" containing the following Prolog code:

```
1 /* 1) Fatos */
2
3
4 /* 2) Regras */
5 nao_vota(X) :-
6   (X < 16; X > 70),
7   write(' não vota').
```
- Results Area:** Three yellow callout boxes on the left point to specific results in the main pane:
  - "Resultado para 15 anos." points to the output for the query `?- /* 3) Consultas */ write('15 anos'), nao_vota(15), nl.`. The output is "15 anos não vota" followed by "true".
  - "Resultado para 25 anos." points to the output for the query `?- /* 3) Consultas */ write('25 anos'), nao_vota(25), nl.`. The output is "25 anosfalse".
  - "Resultado para 75 anos." points to the output for the query `?- /* 3) Consultas */ write('75 anos'), nao_vota(75), nl.`. The output is "75 anos não vota" followed by "true".
- Bottom Navigation:** Examples, History, Solutions, and Run! buttons.



# SWISH - Prolog

- <https://swish.swi-prolog.org/>

/\* 1) Fatos \*/

/\* 3) Consultas \*/

incrementar(3, 5).

/\* 2) Regras \*/

maior(X, Y) :- X > Y.

incrementar(Min, Max) :-

    write(Min),

    nl,

    I is Min + 1,

    not(maior(I, Max)),

    incrementar(I, Max).

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH web-based Prolog environment. At the top, there's a navigation bar with links for File, Edit, Examples, and Help. On the right side of the header, there's a search bar, a user count of "378 users online", and several icons for file operations and notifications (with a "25" badge).

The main interface consists of two main panes. The left pane is a code editor titled "Program" containing the following Prolog code:

```
1 /* 1) Fatos */
2
3 /* 2) Regras */
4 maior(X, Y) :- X > Y.
5 incrementar(Min, Max) :-
6   write(Min),
7   nl,
8   I is Min + 1,
9   not(maior(I, Max)),
10  incrementar(I, Max).
11
```

The right pane is a terminal window showing the execution of a query:

```
?- /* 3) Consultas */ incrementar(3, 5).  
3  
4  
5  
false
```

At the bottom of the terminal window, the query is repeated:

```
?- /* 3) Consultas */  
incrementar(3, 5).
```

At the very bottom of the screen, there are links for Examples, History, and Solutions, along with a checkbox for "table results" and a "Run!" button.



# SWISH - Prolog

- <https://swish.swi-prolog.org/>

/\* 1) Fatos \*/

/\* 3) Consultas \*/

decrementar(5, 3).

/\* 2) Regras \*/

menor(X, Y) :- X < Y.

decrementar(Max, Min) :-

    write(Max),

    nl,

    I is Max - 1,

    not(menor(I, Min)),

    decrementar(I, Min).

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH web-based Prolog environment. The top navigation bar includes links for File, Edit, Examples, Help, and a search bar. A status bar indicates "383 users online". The main area has tabs for Program (selected), Examples, History, and Solutions. The Program tab displays the following Prolog code:

```
1 /* 1) Fatos */
2
3 /* 2) Regras */
4 menor(X, Y) :- X < Y.
5 decrementar(Max, Min) :-
6   write(Max),
7   nl,
8   I is Max - 1,
9   not(menor(I, Min)),
10  decrementar(I, Min).
11
```

The execution results are shown in two panes. The top pane shows the output of a query:

```
/* 3) Consultas */ decrementar(5, 3).
```

Output:

```
5
4
3
false
```

The bottom pane shows the current query being run:

```
?- /* 3) Consultas */ decrementar(5, 3).|
```

At the bottom are buttons for Examples, History, Solutions, and Run, along with a checkbox for "table results".

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

/\* 1) Fatos \*/

/\* 2) Regras \*/

/\* 3) Consultas \*/

```
write('Informe um número: '),
read(Num),
```

A is Num + 2,

B is Num - 2,

C is Num \* 2,

D is Num / 2,

E is Num div 2,

F is Num mod 2,

G is Num \* Num \* Num,

H is Num ^ 2,

```
write('Cubo de '), write(Num), write(': '), write(G), nl, nl,
write('Quadrado de '), write(Num), write(': '), write(H), nl, nl.
```

# SWISH - Prolog

- <https://swish.swi-prolog.org/>

The screenshot shows the SWISH web-based Prolog environment. The top navigation bar includes links for File, Edit, Examples, Help, a user count of 372 users online, a search bar, and various icons for help and documentation.

The main interface consists of two panes:

- Code Editor (Left):** Displays the following Prolog code:

```
1 /* 1) Fatos */
2
3 /* 2) Regras */
4
```
- Interactive Console (Right):** Shows the following interaction:

Informe um número:  
5  
Cubo de 5: 125  
  
Quadrado de 5: 25  
  
A = 7,  
B = 3,  
C = 10,  
D = 25

?- /\* 3) Consultas \*/  
write('Informe um número: '),  
read(Num),  
  
A is Num + 2,  
B is Num - 2,  
C is Num \* 2,  
D is Num / 2,  
E is Num div 2,  
F is Num mod 2,  
G is Num \* Num \* Num,  
H is Num ^ 2,  
  
write('Cubo de '), write(Num), write(': '), write(G), nl, nl,  
write('Quadrado de '), write(Num), write(': '), write(H), nl, nl.

# Exercícios

- 1) Considerando o código ao lado, realize as seguintes consultas em Prolog:
- a) Listar todas as mães.
  - b) Listar todas as filhas.
  - c) Listar todos os filhos.
  - d) Listar as avós.
  - e) Listar os avôs.
  - f) Listar os netos.
  - g) Listar as netas.
  - h) Listar todas as irmãs.
  - i) Listar todos os irmãos.
  - j) Listar os filhos de Maria.
  - k) Listar as filhas de Sílvia.
  - l) Listar os filhos de rodrigo.

```
/* 1) Fatos */
progenitor(maria, rodrigo).      /* Predicado binário */
progenitor(emmanuel, rodrigo).

progenitor(rodrigo, julia).
progenitor(silvia, julia).

progenitor(rodrigo, jorge).
progenitor(ana, jorge).

masculino(emmanuel). /* Predicado unário */
masculino(rodrigo).
masculino(jorge).

feminino(maria).
feminino(ana).
feminino(julia).
feminino(silvia).

/* 2) Regras */
irmao(X, Y) :- progenitor(Z, X), progenitor(Z, Y), masculino(X).
irma(X, Y) :- progenitor(Z, X), progenitor(Z, Y), feminino(X).

pai(X, Y) :- progenitor(X, Y), masculino(X).
mae(X, Y) :- progenitor(X, Y), feminino(X).

/* 3) Consultas */
?- mae(ana, jorge).
```

# Exercícios

- 2) Considerando o código ao lado, realize a seguintes consultas em Prolog:
- Listar todas as classes que herdam da classe Pessoa.
  - Listar todas as classes que herdam da classe Aluno.
  - Listar todas as classes que herdam da classe Professor.
  - Listar todas as classes que contratam a interface Pessoa.
  - Listar todas as classes que contratam a interface Professor.
  - Listar todas as classes filhas da classe Pessoa.
  - Listar todas as classes filhas da classe Professor.
  - Listar a interface que a classe Aluno implementa

```
interface("ItfPessoa").  
interface("ItfAluno").  
interface("ItfProfessor").  
interface("ItfCoordenador").  
  
classe("ClsPessoa").  
classe("ClsAluno").  
classe("ClsProfessor").  
classe("ClsCoordenador").  
  
implementa("ClsPessoa", "ItfPessoa").  
implementa("ClsAluno", "ItfAluno").  
implementa("ClsProfessor", "ItfProfessor").  
implementa("ClsCoordenador", "ItfCoordenador").  
  
herda("ClsAluno", "ClsPessoa").  
herda("ClsProfessor", "ClsPessoa").  
herda("ClsCoordenador", "ClsProfessor").  
  
subclasse(X, Y) :- classe(X), herda(X, Y), classe(Y).  
contrata(X, Y) :- classe(X), implementa(X, Y), interface(Y).
```



# Exercícios

- 3) Em Prolog, apresente, no mínimo, 3 fatos, 3 regras e 5 consultas para os seguintes domínios:
- a) Aviação.
  - b) Comércio.
  - c) Medicina.
  - d) Meio ambiente.
  - e) Tecnologia.
- 