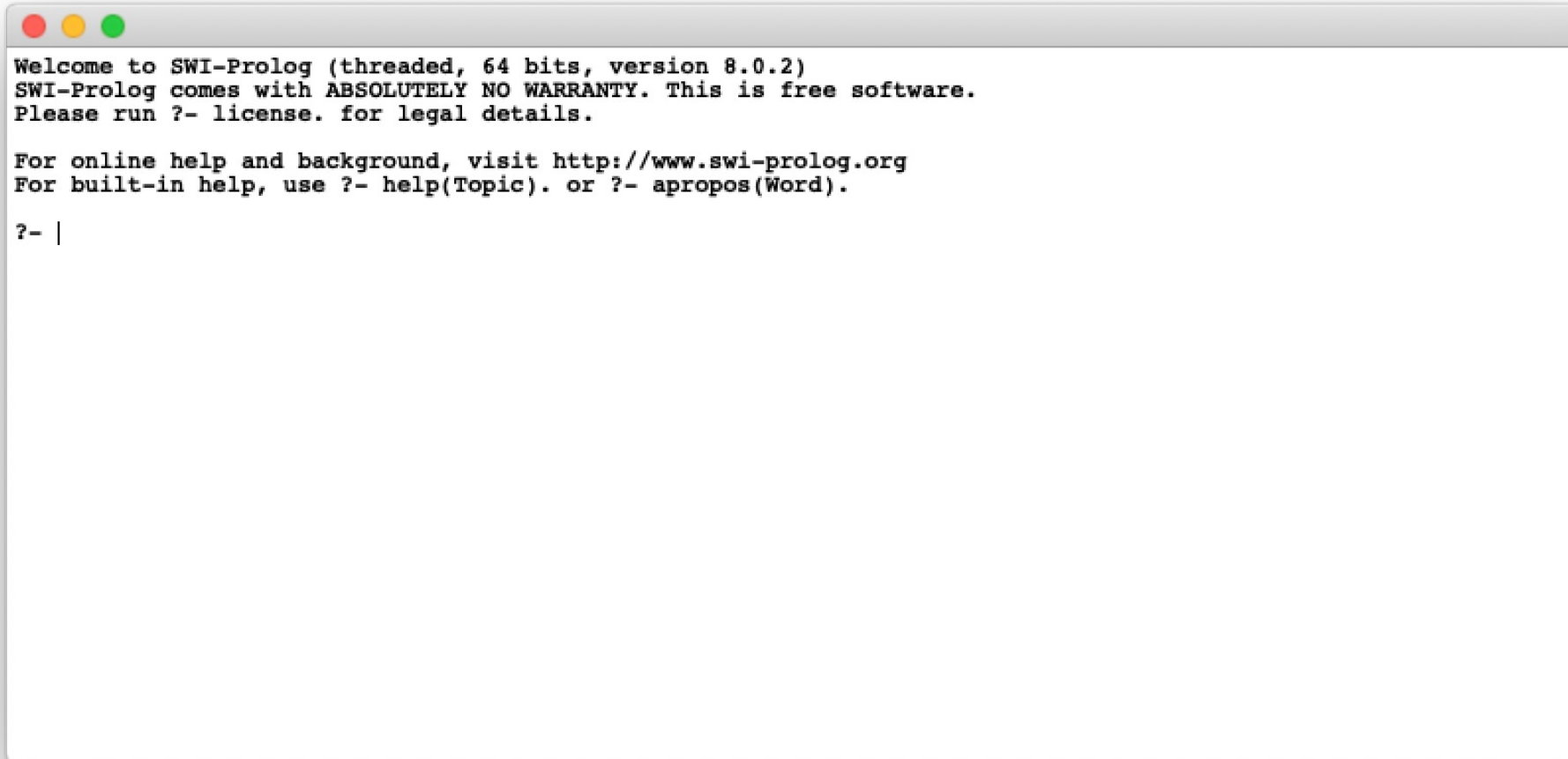


Inteligência Artificial

Programação em Lógica I – TP

Licenciatura em Engenharia Informática
2024/2025

<https://www.swi-prolog.org/Download.html>

A screenshot of a SWI-Prolog terminal window. The window has a standard macOS-style title bar with red, yellow, and green window control buttons. The text inside the window is as follows:

```
Welcome to SWI-Prolog (threaded, 64 bits, version 8.0.2)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit http://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- |
```

teste.pl UNREGISTERED

teste.pl

```
1
2 hello(world).
3
4 ai_is_cool.
5
6 uc('inteligencia artificial', lei).
7 uc('fundamentos de programação', lei).
8 uc('tecnologias escaláveis para análise de dados', mei).
9
10
```

Line 9, Column 1

```
Welcome to SWI-Prolog (threaded, 64 bits, version 8.0.2)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit http://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- consult(['Users/davidecarneiro/Downloads/teste.pl']).
true.

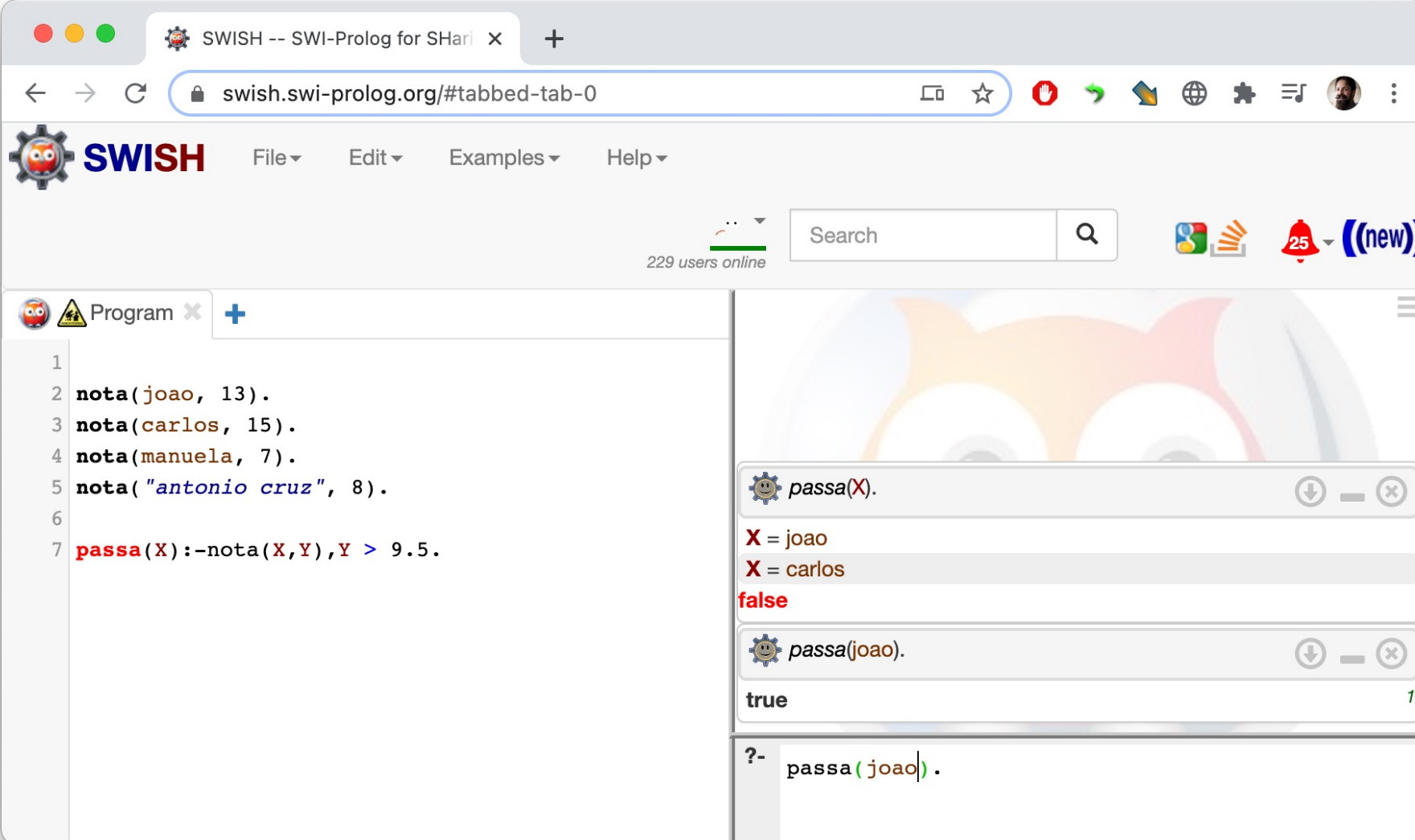
?- hello(X).
X = world.
|
?- ai_is_cool.
true.

?- ai_is_not_cool.
ERROR: Undefined procedure: ai_is_not_cool/0 (DWIM could not correct goal)
?- uc(X, lei).
X = 'inteligencia artificial'
X = 'fundamentos de programação'.

?- uc('inteligencia artificial', mei).
false.

?-
```

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



SWISH -- SWI-Prolog for SHari

swish.swi-prolog.org/#tabbed-tab-0

SWISH File Edit Examples Help

229 users online

Search

Program

```
1
2 nota(joao, 13).
3 nota(carlos, 15).
4 nota(manuela, 7).
5 nota("antonio cruz", 8).
6
7 passa(X):-nota(X,Y),Y > 9.5.
```

passa(X).

X = joao
X = carlos
false

passa(joao).

true

?- passa(joao).

_conceitos básicos

Factos: correspondem a axiomas;

```
rio(douro).  
pai(pedro, ana).
```

A abordar na próxima
aula

Regras: correspondem a implicações;

```
neto(N,A):- filho(N,P),(descendente(P,A,_);descendente(P,_,A)).
```

Questões: permitem interrogar a Base de Conhecimento (BC)

```
?-pai(P,ana).  
?-pai(pedro,ana).  
?-neto(rui,A).
```

_factos

Tem um functor (nome, começado por minúscula);

Tem **zero ou mais** argumentos, englobados por parêntesis e separados por vírgulas, e termina com um **ponto**;

Argumentos são:

- Átomos - valores constantes, números, strings...;
- Variáveis - começadas por uma maiúscula.

_factos

casados(rui, isabel).

Functor: casados

Argumentos: 2 átomos – rui e isabel

ligado.

Functor: ligado

Argumentos: nenhum

potência(X,0,1).

Functor: potência

Argumentos: 3 – uma variável e 2 valores numéricos

_uma base de conhecimento

fica(porto,portugal).
fica(lisboa,portugal).
fica(coimbra,portugal).
fica(caminha,portugal).
fica(madrid,espanha).
fica(barcelona,espanha).
fica(zamora,espanha).
fica(orense,espanha).
fica(toledo,espanha).

(Continuação)

passa(douro,porto).
passa(douro,zamora).
passa(tejo,lisboa).
passa(tejo,toledo).
passa(minho,caminha).
passa(minho,orense).

_questões sobre a BC

fica(porto,portugal).

falha

fica(lisboa,portugal).

falha

fica(coimbra,portugal).

falha

fica(caminha,portugal).

falha

fica(madrid,espanha).

sucesso

fica(barcelona,espanha).

fica(zamora,espanha).

fica(orense,espanha).

fica(toledo,espanha).

?-fica(madrid,espanha).

yes

passa(douro,porto).

falha

passa(douro,zamora).

falha

passa(tejo,lisboa).

falha

passa(tejo,toledo).

falha

passa(minho,caminha).

falha

passa(minho,orense).

falha

?-passa(mondego,coimbra).

no

_questões com variáveis

fica(porto,portugal).

fica(lisboa,portugal).

fica(coimbra,portugal).

fica(caminha,portugal).

fica(madrid,espanha).

fica(barcelona,espanha).

fica(zamora,espanha).

fica(orense,espanha).

fica(toledo,espanha).

passa(douro,porto).

passa(douro,zamora).

passa(tejo,lisboa).

passa(tejo,toledo).

passa(minho,caminha).

passa(minho,orense).

falha

falha

falha

falha

sucesso

falha

falha

sucesso

?-fica(X,espanha).

X=madrid <cr>

yes

?-fica(X,Y).

X=porto Y=portugal <cr>

yes

?- passa(tejo,C).

C=lisboa <cr>

yes

_alternativas

fica(porto,portugal).
fica(lisboa,portugal).
fica(coimbra,portugal).
fica(caminha,portugal).
fica(madrid,espanha).
fica(barcelona,espanha).
fica(zamora,espanha).
fica(orense,espanha).
fica(toledo,espanha).

passa(douro,porto).
passa(douro,zamora).
passa(tejo,lisboa).
passa(tejo,toledo).
passa(minho,caminha).
passa(minho,orense).

falha
falha
falha
falha
sucesso
sucesso
sucesso
sucesso
sucesso

?-fica(X,espanha).
X=madrid ;
X=barcelona ;
X=zamora ;
X=orense ;
X=toledo
yes

_guardar alternativas numa lista

fica(porto,portugal).
fica(lisboa,portugal).
fica(coimbra,portugal).
fica(caminha,portugal).
fica(madrid,espanha).
fica(barcelona,espanha).
fica(zamora,espanha).
fica(orense,espanha).
fica(toledo,espanha).

falha
falha
falha
falha
sucesso
sucesso
sucesso
sucesso
sucesso

?-findall(X,fica(X,espanha),L).
L=[madrid,barcelona,zamora,orense,toledo]
yes

(a abordar futuramente)

_conjunção

fica(porto,portugal). **sucesso**

fica(lisboa,portugal).

fica(coimbra,portugal).

fica(caminha,portugal).

fica(madrid,espanha).

fica(barcelona,espanha).

fica(zamora,espanha).

fica(orense,espanha).

fica(toledo,espanha).

passa(douro,porto). **sucesso**

passa(douro,zamora).

passa(tejo,lisboa).

passa(tejo,toledo).

passa(minho,caminha).

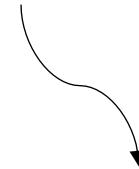
passa(minho,orense).

?-fica(X,portugal),passa(R,X).

X=porto R=douro <cr>

yes

E se fossem pedidas
alternativas com o ; ?



?-fica(X,portugal),passa(R,X).

X=porto R=douro ;

X=lisboa R=tejo ;

X=caminha R=minho

_disjunção

fica(porto,portugal). **sucesso**

fica(lisboa,portugal).

fica(coimbra,portugal).

fica(caminha,portugal).

fica(madrid,espanha).

fica(barcelona,espanha).

fica(zamora,espanha).

fica(orense,espanha).

fica(toledo,espanha).

passa(douro,porto).

passa(douro,zamora).

passa(tejo,lisboa).

passa(tejo,toledo).

passa(minho,caminha).

passa(minho,orense).

?-fica(X,portugal);fica(X,espanha).

X=porto <cr>

yes

?-fica(X,portugal);fica(X,espanha).

X=porto ;

X=lisboa ;

X=coimbra ;

X=caminha ;

X=madrid ;

X=barcelona ;

X=zamora ;

X=orense ;

X=toledo

yes

E se fossem pedidas
alternativas com o ; ?

_negação

fica(porto,portugal).
fica(lisboa,portugal).
fica(coimbra,portugal).
fica(caminha,portugal).
fica(madrid,espanha).
fica(barcelona,espanha).
fica(zamora,espanha).
fica(orense,espanha).
fica(toledo,espanha).

sucesso

passa(douro,porto).
passa(douro,zamora).
passa(tejo,lisboa).
passa(tejo,toledo).
passa(minho,caminha).
passa(minho,orense).

falha
falha
falha
falha
falha
falha

?-not(fica(porto,portugal)).

No

?-not(passa(mondego,coimbra)).

yes

Inteligência Artificial

Programação em Lógica I – TP

Licenciatura em Engenharia Informática
2024/2025