Guidelines for using SWI-Prolog on login.cpp.edu

1. Write a Prolog program

Use any text editor such as pico, nano, …; save it as a .pl file, e.g. lab1.pl

1. Load the program

%swipl –s lab1.pl

Or launch prolog compiler first and then load the file using consult

%swipl

?-[‘lab1.pl’].

Or

?-consult(‘lab1.pl’).

(don’t forget the . at the end)

To exit Prolog, use

?- halt.

1. Prolog basics
2. Case sensitive; functions such as sqrt, mod use lower-case; variables use uppercase such as X, Y, Z, N1, N2, …
3. Arithmetic operators:+, -, \*, /, …
4. Comparison operators:

|  |  |
| --- | --- |
| **Arithmetic examples** | **Prolog Notation** |
| x<y | X < Y. |
| x\le y | X =< Y. |
| x=y | X =:= Y. |
| x\not=y | X =\= Y. |
| x\ge y | X >= Y |
| x>y | X > Y |

1. = vs is

For example:

?- 7 = 5 + 2.

false.

?- 7 is 5 + 2.

true.

**=** is unification, not assignment, for example

?- A + B = 5 + 2.

A = 5,

B = 2

Use **is** for assignment

1. Sample programs
2. Traditional arithmetic quad.pl (a simplified version of PA2 problem.)
3. Recursive function: factorial.pl
4. PA4: count numbers (count.pl) (note: atom(x): x not number, not list)

?- count([],S).

S = 0 .

?- count(a,S).

S = 0.

?- count(3,S).

S = 1 .

?- count([5,1,a,4]).

?- count([5,1,a,4], S).

S = 3 .

?- count([[1,1,[a]],[7,[3,[b]]]], S).

S = 4 .

?- halt.

1. Quiz 2 color count problem (note: treat b, r, g as atoms, not strings.)