Guide 12 – Logic

* Propositions:
  + a statement or assertion that expresses a judgement or opinion.
  + True or false.
  + Compound propositions formed by connecting individual propositions with logical connectives.
  + Propositions without logical connectives are called atomic propositions.
  + Foundation of first-order logic and higher-order logic.
  + Sometimes called zeroth-order logic.
* Logical inference:
  + Deductive inference.
  + Conclusion always follows the stated premises.
  + If the premise is true, then the conclusion is valid.
  + Three stages:
    - Person must understand the meaning of the premises.
    - Person must be able to formulate a valid conclusion.
    - Person should evaluate their conclusion to test its validity.
* If term1 and term2 are constants, then term1 and term2 unify if and only if they are the same atom, or the same number.
* If term1 is a variable and term2 is any type of term, then term1 and term2 unify, and term1 is instantiated to term2 . Similarly, if term2 is a variable and term1 is any type of term, then term1 and term2 unify, and term2 is instantiated to term1 . (So if they are both variables, they’re both instantiated to each other, and we say that they share values.)
* If term1 and term2 are complex terms, then they unify if and only if:
  + They have the same functor and arity, and
  + all their corresponding arguments unify, and
  + the variable instantiations are compatible. (For example, it is not possible to instantiate variable X to mia when unifying one pair of arguments, and to instantiate X to vincent when unifying another pair of arguments .)
* Two terms unify if and only if it follows from the previous three clauses that they unify.

Pessimism is safe but optimism is a lot faster.

SWI-Prolog does not do the occurs check first.

Command: unify\_with\_occurs\_check/2. (performs occurs check)