Chapter 1 - Tutorial Introduction

* 1.1 Prolog
  + Used for applications of symbolic computation, including:
    - Relational databases
    - Mathematical logic
    - Abstract problem solving
    - Understanding natural language
    - Design automation
    - Symbolic equation solving
    - Biochemical structure analysis
    - Many areas of artificial intelligence
* 1.2 Objects and Relationships
  + Prolog is used for solving problems that involve objects and the relationships between objects
* 1.3 Programming
  + Specifying some facts about objects and their relationships
  + Defining some rules about objects and their relationships
  + Asking questions about objects and their relationships
* 1.4 Facts
  + The names of all relationships and objects must begin with a lower-case letter
  + The relationship is written first, and the objects are written separated by commas, and the objects are enclosed by a pair of round brackets
  + The dot character “.” Must come at the end of a fact. The dot is what some people also called a “period” or a “full stop”
    - Examples:
      * valuable(gold).
      * female(jane).
      * owns(jane, gold).
      * father(john, mary).
      * gives(john, book, mary).
* 1.5 Questions
  + ?- owns(mary, book).
  + ?- likes(joe, money).
    - no
  + ?- likes(mary, book).
    - yes
* 1.6 Variables
  + likes(john, flowers)
  + ?- likes(john, X)
    - X = flowers
* 1.7 Conjunctions
  + ?- likes(john, mary), likes(mary, john).
    - “Does John like Mary and does Mary like John?”
  + Backtracking behavior in Prolog
    - Database is searched for the first goal
    - The database is searched
    - The marked place is found, so Prolog begins searching from after the fact
    - Prolog now tries the second goal
    - Both goals have been finished (in a conjunction)
* 1.8 Rules
  + ?- sister\_of(alice, edward). When asked a question given this database and rule for sister\_of, Prolog proceeds as follows:
    - Question unifies with the head of the only sister\_of rule above, so X in the rule become instantiated to alice, and I becomes instantiated to edward
    - First goal is female(alice) because X was instantiated to alice in the previous step
    - Prolog searches for parents(alice, M, F)., where M and F will unify with any arguments because they are not instantiated
    - Prolog searches for parents(edward, victoria, albert) because Y is known as edward from the question, and M and F were known to stand for victoria and albert from the previous goal
* 1.9 Summary
  + Asserting facts about objects
  + Asking questions about the facts
  + Using variables and what their scopes are
  + Conjunction as a way of saying “and”
  + Representing relationships in the form of rules
  + An introduction to backtracking