Concepts of programming languages

Prolog

Winand, Roald, Sjoerd, Alexey and Anvar

Universiteit Utrecht

[Faculty of Science
Information and Computing
Sciences]

Terms

- building blocks of facts, rules, and queries.
- 4 kinds of terms:
- atoms
- numbers (both are called constants)
- variables
- complex terms

Atom

Either:

- string of charachrers..
- arbitraty string of ch. in ” ”
- string of special characters

Numbers

- Floats
- Integers
- Straigtforward syntax

Variable

- starts with upper-case letter or _
- anonimous variable _

Complex term

- building block: functor
- nested functors make up complex terms

Examples

Clauses

- Rules (clauses) state information that is conditionally true of the situation of interest.
- term1 :- term2
- term1 is true if term2 is true.

some Examples again

$\ldots \_\_\_\_$

Unification (how it works)

Two terms unify if they are the same term or if they contain variables that can be uniformly instantiated with terms in such a way that the resulting terms are equal.

what this means??

…examples

more on unification..

- two terms either unify or not
- if they unify, we are interested to know how the variables have to be instantiated to make the terms unify.

more precise rules:

Two terms (term1 and term2) unify:

- If they are both constants, they unify iff they are the same atom (or number)
- If term1 is a variable and term2 is any term, then they unify and term1 is instantiated to term2.
- If both terms are variables, they're both instantiated to each other.
- If both are complex terms and … (next silde)
- Iff it follows from the rules above that they unify.

Some examples first

Some examples first...

If term1 and term2 are complex terms, they unify iff:

- they have the same functor and arity (nr. of args)
- all their corr. args unify
- the variable instantiations are compatible

Universiteit Utrecht

[Faculty of Science
Information and Computing
Sciences]