

Integrated Logic Systems (Part I)

Prof. Michael Schroeder, Sebastian Voigt

International Master Program in Computational Logic — summer term 2010

04.05.2010

Exercise 2.1

Consider the special instructions for constants (Slides Prolog(I)/28-29) and lists (Slide Prolog(I)/31). Compile the \mathcal{L}_1 query $? - l([a, s(s(Y))|Y], c)$ and \mathcal{L}_1 program $l([X, Y, X, b], c)$. with respect to these instructions, execute the resulting code and give the CAS.

Exercise 2.2

The goal of this exercise is to extend the WAM such that it is able to handle integer arithmetic with the operations addition, subtraction and multiplication, denoted by the operators $+$, $-$ and $*$, respectively. The operators can be written in infix notation with the usual precedences and are left-associative. For example, $3 * 4 - 2 * 8 + 7$ is interpreted as $((3 * 4) - (2 * 8)) + 7$. The extension of the WAM is supposed to provide the following two predicates:

$==/2$: $expr_1 == expr_2$ succeeds for two arithmetic expressions $expr_1$ and $expr_2$ iff they can be evaluated to the same integer value

$is/2$: $X is expr$ tries to unify X with the result of evaluating the arithmetic expression $expr$, which succeeds iff X is unbound or a number equal to the evaluated value

- Define a function `evaluate(a:address):integer` which processes the heap representation of a term starting at address `a` and returns its integer value. You may assume functions `+`, `-` and `*` for the addition, subtraction and multiplication of two integer values, respectively.
- Define two new machine instructions `call ==/2` and `call is/2` which realize the predicates as specified above.
- Give the \mathcal{M}_2 code for query $? - X is 3, 8 == 3 + 4 * X - 7$. and trace its execution.

Exercise 2.3

- (*Last call optimization*) Permanent variables (Slide Prolog(II)/6) need no longer be remembered after all put instructions for the ultimate `call` in the body of a rule are passed. Thus an environment can be discarded before this `call` in order to save space on the stack.

Properly change the WAM to achieve the mentioned effect and test it with the \mathcal{L}_2 query $? - p(a, X), q(X)$. and the \mathcal{L}_2 program

$$\begin{aligned} p(X, f(X)) & : - q(f(f(X))), r(X). \\ q(X) & : - r(X). \\ r(X) & . \end{aligned}$$

- (*Chain rules*) Chain rules are rules that have exactly one body atom. In which cases can we omit environments for chain rules? And for queries?