Formal Specification of Abstract Datatypes
Exercises 2+3 (May 12+26)

Wolfgang Schreiner
Wolfgang.Schreiner@risc.uni-linz.ac.at

April 7, 2008

The result for each exercise is to be submitted by the deadline stated above via the Moodle interface as a .zip or .tgz file which contains

- a PDF file with
  - a cover page with the title of the course, your name, Matrikelnummer, and email-address,
  - the content required by the exercise (specification, source, proof),
- (if required) the CafeOBJ (.mod) file(s) of the specifications.

Exercise 2: Loose Specifiction of Integers

Assume you are given a strictly adequate specification of the classical algebra of Boolean values by a sort *bool* with free constructors $True :\rightarrow bool$, $False :\rightarrow bool$ and the logical operations $not : bool \rightarrow bool$, $and : bool \times bool \rightarrow bool$, $or : bool \times bool \rightarrow bool$.

Furthermore, assume you are also given a strictly adequate specification of the classical algebra of natural numbers by a sort *nat* with free constructors $0_n :\rightarrow nat$ and $succ_n : nat \rightarrow nat$, constant $1_n :\rightarrow nat$, and operations $-_n : nat \rightarrow nat$ ($x -_n y = 0$ for $x \leq y$), $+_n : nat \times nat \rightarrow nat$, $*_n : nat \times nat \rightarrow nat$, $=_n : nat \times nat \rightarrow bool$, $<_n : nat \times nat \rightarrow bool$.

With the help of these sorts and operations, write in a logic of your choice (please state explicitly which one you coose) a loose specification (potentially with constructors) of a sort *int* with constants $0_i :\rightarrow int$, $1_i :\rightarrow int$ and operations $-_i : int \rightarrow int$, $+_i : int \times int \rightarrow int$, $*_i : int \times int \rightarrow int$, $=_i : int \times int \rightarrow bool$, $<_i : nat \times nat \rightarrow bool$. This specification shall be strictly adequate with respect to the classical algebra of integer numbers.

For writing the specification, think how you can construct a representation of the integer numbers as "natural numbers with a sign" and define corresponding

constructors. Based on this representation, write (potentially recursive) definitions of the other constants and operations.

Also develop an executable version of this specification in CafeOBJ (i.e. a tight module `module! MYINT {...}` based on the existing modules `BOOL` and `NAT`) and test it with a couple of sample reductions. In particular, check whether

$$-_i(0_i) =_i 1_i +_i -_i(1_i)$$

reduces to *true*.

Exercise 3: Strict Adequacy of Specification

Prove that above specification of the integers is strictly adequate, i.e. that every algebra satisfying this specification is isomorphic to the "classical" integer algebra with carrier $\mathbb{Z}$ (you only need to show the homomorphism condition for the operations 0_i, 1_i $-_i$, $+_i$, and $=_i$).

For performing the proof, you have to define (as shown in class) a unique term representation for every element of $\mathbb{Z}$ (including 0).