Formal Specification of Abstract Datatypes
Exercise 5 (July 2)

Wolfgang Schreiner
Wolfgang.Schreiner@risc.jku.at

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 5: Strict Adequacy of Queues

Take the initial specification of `Queue` from Exercise 4 (where `Elem` is replaced by a sort `Nat` with free constructors `0` $:\to$ `Nat`, `s` : `Nat` $\to$ `Nat` which can be assumed to strictly adequately specify the algebra of natural numbers). Show that this specification is strictly adequate with respect to the classical algebra of queues using the proof technique of characteristic term algebras.

1. Define a characteristic term algebra for `Queue` and prove that it is indeed characteristic.
2. With the help of the characteristic term algebra, prove the strict adequacy of `Queue` with respect to the algebra of queues.

Hints: the classical algebra of queues can be defined as follows:

1. The carrier *Queue* is defined as

   $Queue := \bigcup_{n\in\mathbb{N}} Q_n$
   $Q_n := \mathbb{N}_n \to \mathbb{N}$

where $\mathbb{N}_n$ is the set of the first n natural numbers. In other words, Q_n is the set of finite sequences (of natural numbers) of length n and *Queue* is the set of all finite sequences. Consequently, for every sequence $q \in Q_n$ and position $i \in \mathbb{N}_n$, the term $q(i)$ denotes the element at position i of sequence q.

2. The *Queue* functions are defined as

$$
\begin{aligned}
&\mathit{empty} : Q_0, \mathit{empty} := \emptyset \\
&\mathit{isempty} : \mathit{Queue} \to \mathbb{B} \\
&\mathit{isempty}(q) := \text{if } q \in Q_0 \text{ then } \mathit{true} \text{ else } \mathit{false} \\
&\mathit{enqueue} : \mathit{Queue} \times \mathbb{N} \to \mathit{Queue} \\
&\mathit{enqueue}(q, e) := \\
&\quad \text{let } n = \text{such } i \in \mathbb{N} : q \in Q_i \\
&\quad \text{such } q' \in Q_{n+1} : \forall i \in \mathbb{N}_{n+1} : q'(i) = \text{if } i < n \text{ then } q(i) \text{ else } e \\
&\mathit{dequeue} : \mathit{Queue} \to \mathit{Queue} \\
&\mathit{dequeue}(q) := \\
&\quad \text{let } n = \text{such } i \in \mathbb{N} : q \in Q_i \\
&\quad \text{such } q' \in Q_{n-1} : \forall i \in \mathbb{N}_{n-1} : q'(i) = q(i+1) \\
&\mathit{head} : \mathit{Queue} \to \mathbb{N} \\
&\mathit{head}(q) := q(0)
\end{aligned}
$$

As for proving properties of *Queue*: in order to show that, for some predicate P, the formula $\forall q \in \mathit{Queue} : P(q)$ is true, it suffices to show that $\forall n \in \mathbb{N} : \forall q \in Q_n : P(q)$ is true (which can be shown by induction on n).