Software Modeling and Verification
Lehrstuhl für Informatik 2
RWTH Aachen University

Prof. Dr. Ir. J.-P. Katoen
Priv.-Doz. Dr. T. Noll
T. Han, M. Neuhäußer

Modeling Concurrent and Probabilistic Systems

Winter Term 07/08

– Series 3 –

Hand in until November 09 before the exercise class.

Exercise 1 (4 + 1 points)

In analogy to the buffer, this exercise deals with the *semaphore* data structure which can be regarded as a buffer for identical entries.

a) An n-ary semaphore ($n \geq 1$) is given by the following specification:

$$\begin{aligned} Sem_0^n(get, put) &= get.Sem_1^n(get, put) \\ Sem_k^n(get, put) &= get.Sem_{k+1}^n(get, put) + put.Sem_{k-1}^n(get, put) \qquad (0 < k < n) \\ Sem_n^n(get, put) &= put.Sem_{n-1}^n(get, put). \end{aligned}$$

Prove that it can also be specified by composing n parallel instances of a unary semaphore:

$$\begin{aligned} S^n(get, put) &= \underbrace{S_0(get, put) \mid\mid S_0(get, put) \mid\mid \cdots \mid\mid S_0(get, put)}_{n \text{ times}} \\ S_0(get, put) &= get.S_1(get, put) \\ S_1(get, put) &= put.S_0(get, put). \end{aligned}$$

To this aim, show that $Sem_0^n \sim S^n$ by constructing an appropriate strong bisimulation relation.

b) Show that the two variants of the two place buffer given in the lecture are not strongly bisimilar.

Exercise 2 (4 points)

Prove or disprove the following strong bisimulation equivalences between process terms where $P, Q, R \in Proc$ and $a, b \in A$:

a) $P + \mathsf{nil} \sim P$

b) $P + P \sim P$

c) $P \mid\mid (Q \mid\mid R) \sim (P \mid\mid Q) \mid\mid R$

d) $P \mid\mid P \sim P$

Exercise 3 (3 points)

a) Show that the car locking system (cf. Exercise 2.1) as presented in the exercise class has a deadlock.

b) Develop a deadlock free specification of the system.

Exercise 4 **(1+3 points)**

In order to enrich the CCS by a sequential composition operator, we define *well terminating* processes: A process $P \in Proc$ is *well terminating* if for every $P' \in Proc$ with $P \rightarrow^* P'$,

- $P' \xrightarrow{done}$ is not possible and
- if $P' \xrightarrow{\overline{done}}$, then $P' \sim \overline{done}.\mathsf{nil}$.

In other words, the output action $\overline{done}$ signals the (successful) termination of P.

a) Show that nil is a well terminating process, and that the set of well terminating processes is closed under action prefixing, nondeterministic choice and restriction.

b) Give appropriate CCS definitions for the following processes and operators:

Done	The process which is able to execute just one $\overline{done}$ action.
Seq	The process P *Seq* Q denotes the sequential execution of P and Q, assuming that P is well terminating. Moreover P *Seq* Q should be well terminating if both P and Q are.
Par	The process P *Par* Q denotes the parallel execution of P and Q. It should be well terminating if both P and Q are.