Foundations of Programming

– Concurrency –

Session 10 – April 18, 2002

Uwe Nestmann

EPFL-LAMP

Session 10: The Scheduler Problem

- informal specification
  - black box
  - flow graph
- specification as sequential process expression
  - transition graph
- implementation as concurrent process expression
  - flow graph
  - transition graph ?
- proof "by hand"
- proof using the CWB

Informal Specification [Mil99, § 3.6]

- processes $P_i, 1 \leq i \leq n$ to be scheduled
- P_i starts by pressing a_i of the scheduler
- P_i completes by signalling b_i to the scheduler
- each P_i *must not* run two tasks at a time
- tasks of different P_i *may* run at the same time
- a_i are required to occur cyclically (initially, 1 starts)
- for each i, a_i and b_i must occur cyclically
- permit maximal "pressure"

Formal Specification [Mil99, § 3.6]

$i \in \{1 \ldots, n\} \quad X \subseteq \{1 \ldots, n\}$

$\mathsf{S}_{i,X}(\vec{a}, \vec{b}) \stackrel{\text{def}}{=}$ scheduler, where i is next and X are running

(* we omit the parameters in the following *)

$$\text{Scheduler} \stackrel{\text{def}}{=} \mathsf{S}_{1,\emptyset}$$

$$\mathsf{S}_{i,X} \stackrel{\text{def}}{=} \begin{cases} \sum_{j\in X} b_j.\mathsf{S}_{i,X-j} & (i \in X) \\ \sum_{j\in X} b_j.\mathsf{S}_{i,X-j} + a_i.\mathsf{S}_{i+1 \bmod n, X\cup i} & (i \notin X) \end{cases}$$

- draw the transition graph for $n = 2$
- show that the scheduler is never deadlocked
- what is the difference when dropping $i \in X$?

Formal “Implementation” (I) [§ 7.3]

$$
\begin{array}{rcl}
A(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & a.c.b.\overline{d}.A \\
\hline
A(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & a.C\langle\,a,b,c,d\,\rangle \\
C(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & c.B\langle\,a,b,c,d\,\rangle \\
B(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & b.D\langle\,a,b,c,d\,\rangle \\
D(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & \overline{d}.A\langle\,a,b,c,d\,\rangle \\
\hline
A_i & \stackrel{\text{def}}{=} & A\langle\,a,b,c_i,c_{i-1}\,\rangle \\
\ldots & & \\
S(\,\vec{a},\vec{b}\,) & \stackrel{\text{def}}{=} & (\boldsymbol{\nu}\vec{c})\;(\;A_1|D_2|\cdots|D_n\;)
\end{array}
$$

$$\boxed{\text{Scheduler} \stackrel{?}{\approx} S\langle\,\vec{a},\vec{b}\,\rangle}$$

Formal “Implementation” (II) [§ 7.3]

$$
\begin{array}{rcl}
A(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & a.c.(b.\overline{d}.A + \overline{d}.b.A) \\
\hline
A(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & a.C\langle\,a,b,c,d\,\rangle \\
C(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & c.E\langle\,a,b,c,d\,\rangle \\
E(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & b.D\langle\,a,b,c,d\,\rangle + \overline{d}.B\langle\,a,b,c,d\,\rangle \\
B(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & b.A\langle\,a,b,c,d\,\rangle \\
D(\,a,b,c,d\,) & \stackrel{\text{def}}{=} & \overline{d}.A\langle\,a,b,c,d\,\rangle \\
\hline
A_i & \stackrel{\text{def}}{=} & A\langle\,a,b,c_i,c_{i-1}\,\rangle \\
\ldots & & \\
S(\,\vec{a},\vec{b}\,) & \stackrel{\text{def}}{=} & (\boldsymbol{\nu}\vec{c})\;(\;A_1|D_2|\cdots|D_n\;)
\end{array}
$$

$$\boxed{\text{Scheduler} \stackrel{?}{\approx} S\langle\,\vec{a},\vec{b}\,\rangle}$$