Ludwig-Maximilians-Universität München
Institut für Informatik
Prof. Dr. Christian Prehofer, Christian Kroiß

SoSe 2011
Übungsblatt 5

Formale Techniken in der Software-Entwicklung

Aufgabe 5-1 **Transitionssysteme und Fairness** Hausaufgabe

Sei T ein Transitionssystem, dass sich aus dem im Folgenden informell beschriebenen Verhalten zweier Prozesse P_1, P_2 und einer Variable $x \in \mathbb{N}$ ergibt:

- P_1 kann mit der Aktion α den Wert von x um 1 erhöhen.
- P_2 kann, wenn x einen geraden Wert hat, mit der Aktion β den Wert von x auf 1 setzen.
- Initial gilt $x = 0$.

a) Illustrieren Sie in einer (endlichen) Zeichnung den Aufbau des Transitionssystems.

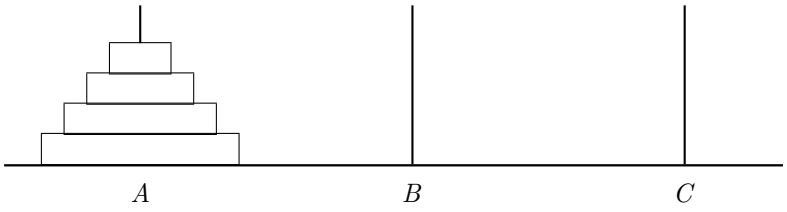
b) Geben Sie einen Ablauf an, der stark fair bzgl. der Aktionen α und β ist.

c) Geben Sie einen Ablauf an, der schwach fair, aber nicht stark fair bzgl. der Aktionen α und β ist.

d) Zeigen Sie: Jeder Ablauf von T ist schwach fair bzgl. der Aktionen α und β.

Aufgabe 5-2 **Türme von Hanoi** Hausaufgabe

Beim Spiel *Türme von Hanoi* soll ein Turm von n Scheiben von A nach C bewegt werden. In jedem Zug wird die oberste Scheibe eines Turms genommen und oben auf einen anderen Turm gelegt; dabei dürfen nur die Plätze A, B und C benutzt werden, und es darf niemals eine größere Scheibe auf eine kleinere zu liegen kommen.



Beschreiben Sie die erlaubten Zugfolgen durch ein Transitionssystem.

Besprechung der Aufgaben in der Übung am 10.06.2011. **Bitte wenden!**

Aufgabe 5-3 **Sicherheits- und Lebendigkeitseigenschaften** Hausaufgabe

Welche der folgenden Eigenschaften (im Kontext des Eisenbahnbeispiels aus der Vorlesung) sind Sicherheits- bzw. Lebendigkeitseigenschaften? Begründen Sie kurz Ihre Antwort.

a) "Wann immer ein Zug auf der Brücke ist, stehen die Signale auf rot".

Formal: P_1 ist die Menge aller Zustands-Aktions-Folgen $s_0 \xrightarrow{A_0} s_1 \ldots$, so dass für alle $i \in \mathbb{N}$ gilt: Ist $s_i(trainW) = \mathsf{onbridge}$ oder $s_i(trainE) = \mathsf{onbridge}$, so gilt $s_i(signalW) = s_i(signalE) = \mathsf{rot}$.

b) "Der Zug $trainW$ bleibt so lange vor einem roten Signal stehen, bis dieses auf grün schaltet."

Formal: P_2 ist die Menge aller Zustands-Aktions-Folgen $s_0 \xrightarrow{A_0} s_1 \ldots$, so dass für alle $i \in \mathbb{N}$ gilt: Ist $s_i(trainW) = \mathsf{atsignal}$ und $s_i(signalW) = \mathsf{rot}$, so gibt es ein $j \geq i$ mit $s_j(signalW) = \mathsf{grün}$, und für alle k mit $i \leq k < j$ gilt $s_k(trainW) = \mathsf{atsignal}$.

c) "Der Zug $trainW$ fährt immer wieder auf die Brücke."

Formal: P_3 ist die Menge aller Zustands-Aktions-Folgen $s_0 \xrightarrow{A_0} s_1 \ldots$, so dass für alle $i \in \mathbb{N}$ ein $j \geq i$ existiert mit $s_j(trainW) = \mathsf{onbridge}$.

d) "Stehen beide Züge am Signal, so darf $trainE$ höchstens einmal auf die Brücke, bevor $trainW$ auf die Brücke fährt."

Formal: P_4 ist die Menge aller Zustands-Aktions-Folgen $s_0 \xrightarrow{A_0} s_1 \ldots$, so dass für alle $i < j < k < l \in \mathbb{N}$ gilt: Ist $s_i(trainW) = s_i(trainE) = \mathsf{atsignal}$ und gelten $s_j(trainE) = \mathsf{onbridge}$, $s_k(trainE) \neq \mathsf{onbridge}$ und $s_l(trainE) = \mathsf{onbridge}$, so gibt es ein m mit $i < m < l$ und $s_m(trainW) = \mathsf{onbridge}$.