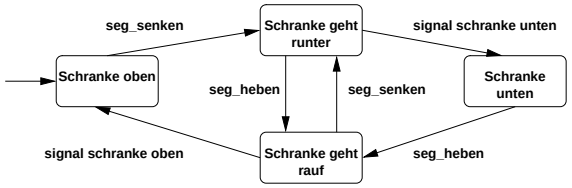
M-INF3 THP2 WiSe 11	**Übung 3** **(Prüfungsvorleistung)**	Prof. Dr. B. Buth 15.11.2010

Punkteverteilung:	**Aufgabe**	**Teilaufgaben**	**Gesamt**
	1	**50+10**	**60**
	2	**40**	**40**
	Gesamt:		**100**
Benötigte Punkte	**60**		

1 CSP Spezifikation

1.1 Prozesse spezifizieren **50 Punkte**

Spezifizieren Sie in CSP ein Prozess-System, das einen Schrankenübergang eines Eisenbahnnetzes modelliert. Der Schrankenübergang besteht aus einem gesicherten Gleisabschnitt, einer zweispurigen Straße und einer Schrankensteuerung mit je zwei Schrankensegmenten für die linke und die rechte Straßenseiten. Der gesicherte Gleisabschnitt kann nur aus einer Richtung befahren werden (vereinfachte Annahme) und wird durch ein Signal für kommende Züge bewacht; das Signal steht auf rot, wenn ein Zug nicht in den Abschnitt einfahren kann, ansonsten auf grün. Die Durchfahrt eines Zuges wird durch zwei weitere Ereignisse für den Eintritt und den Austritt aus dem gesicherten Abschnitt beobachtet.
Ein einzelnes Schrankensegment bewegt sich entsprechend dem folgenden Automaten:



Ein erfolgreiches Durchfahren des Schrankenabschnitts erfolgt in den folgenden Schritten:

- Wenn ein Zug sich dem Übergang nähert (Ereignis zug_kommt), ist das Signal rot. Der Zug sendet eine Durchfahrtsanfrage (df_anfrage) an die Schrankensteuerung.
- Wenn die Schrankensteuerung eine Durchfahrtsanfrage erhält, werden zunächst die rechten Schrankensegmente aus Fahrtrichtung der Fahrspuren gesenkt.
- Erst wenn die rechten Segmente der Schranken unten sind (Signal seg_unten), werden auch die linken Segmente gesenkt.
- Wenn alle vier Segmente unten sind, wird das Durchfahrtssignal für den anfragenden Zug auf grün geschaltet und der Zug erhält die Durchfahrtfreigabe (df_freigabe)
- Bei Einfahrt eines Zuges in den geschützten Gleisabschnitt wird zunächst das Eintrittsereignis (zug_rein) ausgelöst, bei Verlassen des Abschnitts das Austrittsereignis (zug_raus).In der Abstraktion können wir davon ausgehen, dass der Zug diese Ereignisse an die Steuerung sendet.
- Nach dem Eintrittsereignis wird das Signal wieder auf rot gesetzt.
- Wenn nach einem Eintrittsereignis das Austrittsereignis aufgetreten ist, wird das Durchfahrtsignal wieder auf rot gesetzt und anschließend alle Schrankensegmemente gleichzeitig geöffnet.
- Ein nachfolgender Zug kann das Öffnen der Schranke unterbrechen und die Schranke sich wieder schließen lassen - allerdings nur, wenn der vorausfahrende Zug den gesicherten Abschnitt verlassen hat.

Spezifizieren Sie CSP Prozesse für

- die Schrankensegmente entsprechend dem obigen Automaten: SEGMENT (...) =

- die Steuerung: STEUER (...) =
- das Signal: SIGNAL (...) =
- Züge: ZUG (id) =
- den Gesamtprozess des Schrankenübergangs mit 2 Zügen

und verwenden Sie dabei die folgenden Datentypen und Kanäle:

```
nametype ZId = {0,1,2,3}
nametype SegId = {l1,l2,r1,r2} ;; links 1 und 2, rechts 1 und 2
nametype SigStates = {r, g} ;; Signalzustand rot, grün
nametype SegState = {unten, oben, senken, heben };; Segmentzustand
channel zug_kommt, zug_rein, zug_raus: ZID
channel df_anfrage, df_freigabe: ZID ;; Durchfahrt Anfrage und Freigabe
channel seg_senken, seg_heben, seg_unten, seg_oben: SegId
```

Modellieren Sie das Prozesssystem und entwickeln Sie Prüfungen für die folgenden Eigenschaften:

- Züge fahren nur wenn die Schranken alle unten sind
- Kein Zug wird dauerhaft blockiert.

Demonstrieren Sie die Modellierung und die Prüfungen sowie deren Ergebnisse im Praktikumstermin.
Anmerkung: das Spezifikationsskelett findet sich in der Datei `schranken-skelett.csp`

1.2 Verifikation mit Refinement **10 Punkte**

Gegeben seien die folgende CSP-Prozessdefinitionen (machine-readable CSP):

```
1   channel a,b
2
3   N0 = SKIP |~| (a -> N0 [] b -> N1 [] b -> N2)
4   N1 = (a -> N3 [] b -> N2 ) |~| SKIP
5   N2 = (a -> N0 [] b -> N3) |~| SKIP
6   N3 = (a -> N3) [] b -> N3
7
8   X0 = SKIP |~| (a -> X0 [] b -> X1)
9   X1 = SKIP |~| (a -> X0 [] b -> X2)
10  X2 = a -> STOP [] b -> X3
11  X3 = SKIP |~| (a -> X0 [] b -> X2)
```

Beweisen oder widerlegen Sie die folgenden Zusicherungen:

- assert N0 [T= X0
- assert N0 [F= X0
- assert X0 [T= N0
- assert X0 [F= N0

Demonstrieren Sie das Ergebnis im Praktikumstermin

2 CSP Trace Semantik

2.1 Traces für konkrete Prozesse berechnen **40 Punkte**

Gegeben seien die folgenden Prozesse:

$$
\begin{aligned}
P &= (a \rightarrow b \rightarrow \mathit{Skip}) \,\Box\, (b \rightarrow d \rightarrow \mathit{Stop}) \\
Q &= (x \rightarrow \mathit{Skip}) \,\triangle\, (y \rightarrow \mathit{Stop}) \\
R &= (P;\ Q) \setminus \{x, y\} \\
S &= P \underset{\{a,b\}}{\|}_{\{x,y\}} Q
\end{aligned}
$$

Berechnen Sie die Trace-Semantik der Prozesse auf der Basis der Trace-Definitionen im CSP-Guide (Kap. 6). Geben Sie die für die Berechnung verwendeten Regeln an. Referenzversion ist Version 43 des CSP-Guide.

Abgabe: bis Mo, 28.11., 24:00 über EMIL

nur für Aufgabe 2.1!

Die Abgabe der Aufgaben soll in Gruppen von 3-4 Personen erfolgen!