

Betriebssysteme im Wintersemester 2015/2016

Übungsblatt 9

- Abgabetermin:** 21.12.2015, 16:00 Uhr
- Besprechung:** Besprechung der T-Aufgaben in den Tutorien vom 14. – 18. Dezember
Besprechung der H-Aufgaben in den Tutorien vom 21. – 23. Dezember 2015 und vom 07. – 08. Januar 2016
- Ankündigungen:** Am Donnerstag, den 23.06.2016 15:00 Uhr - 16:30 Uhr wird Dr. Ralf Schneider, CIO der Allianz Gruppe, im Rahmen der Vorlesung „Rechnerarchitekturen“ als Industriexperte einen Impulsvortrag halten. Zudem wird es in dieser Zeit Gelegenheit für Fragen geben. Interessierte Studenten sind herzlich eingeladen.

Aufgabe 39: (T) Deadlock Prevention

(– Pkt.)

Eine Methode, um Deadlocks zu vermeiden, ist es, eine der Bedingungen für das Entstehen von Deadlocks im Vorhinein auszuschließen.

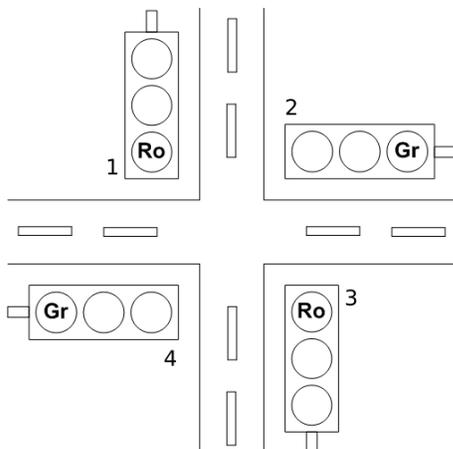
- Geben sie die vier Voraussetzungen für die Entstehung eines Deadlocks an.
- Beschreiben Sie, wie durch eine Ordnung der Ressourcen bei geeigneter Reservierungsstrategie Deadlocks vermieden werden können.

Tipp: Ordnung der Ressourcen bedeutet: Ob ein Prozess ein Betriebsmittel anfordern darf, hängt nicht nur davon ab, ob dieses gerade frei ist, sondern auch davon, welche Betriebsmittel der Prozess zuvor schon angefordert hat. Welche der vier Deadlock-Bedingungen könnte man mit diesem Ansatz ausschließen?

Aufgabe 40: (T) Petrinetze: Modellierung einer Ampelanlage

(– Pkt.)

Wir betrachten im Folgenden die Ampelschaltung einer Kreuzung wie sie hier zu sehen ist.



Gehen Sie von folgenden Randbedingungen aus:

- Zwei sich gegenüberliegende Ampeln schalten ihr Signal stets synchron.
- Eine Ampel schaltet stets in der Reihenfolge **rot** → **rotgelb** → **grün** → **gelb** → **rot** → ...
- Wenn eines der sich gegenüberliegenden Ampelpaare *nicht* auf **rot** steht, dann *muss* das andere Ampelpaars auf rot stehen.

Bearbeiten Sie unter Berücksichtigung der Randbedingungen folgenden Aufgaben:

- a. Modellieren Sie eine einzelne Ampel durch ein Petrinetz. Wählen Sie dabei aussagekräftige Bezeichner für die verschiedenen Stellen Ihres Petrinetzes. Verwenden Sie dabei eine minimale Anzahl an Stellen, Marken, Kanten und Transitionen.
- b. Modellieren Sie nun das Schaltverhalten der beiden Ampeln **1** und **2** (also das von zwei Ampeln, die sich **nicht** gegenüberliegen) durch ein Petrinetz, so dass die genannten Randbedingungen erfüllt sind. Wählen Sie dabei wieder aussagekräftige Bezeichner für die verschiedenen Stellen Ihres Petrinetzes. Verwenden Sie dabei wieder eine minimale Anzahl an Stellen, Marken, Kanten und Transitionen.
- c. Modellieren Sie nun das Schaltverhalten der beiden Ampeln **1** und **3** (also von zwei sich gegenüberliegenden Ampeln) durch ein Petrinetz, so dass die genannten Randbedingungen erfüllt sind. Wählen Sie dabei wieder aussagekräftige Bezeichner für die verschiedenen Stellen Ihres Petrinetzes. Verwenden Sie wieder eine minimale Anzahl an Stellen, Marken, Kanten und Transitionen.
- d. Modellieren Sie nun alle vier Ampeln (also die gesamte Ampelschaltung) durch ein Petrinetz, so dass die genannten Randbedingungen erfüllt sind. Wählen Sie dabei wieder aussagekräftige Bezeichner für die verschiedenen Stellen Ihres Petrinetzes. Verwenden Sie wieder eine minimale Anzahl an Stellen, Marken, Kanten und Transitionen.

Aufgabe 41: (H) Petrinetze: Druckerwarteschlange und Erreichbarkeitsgraph

(16 Pkt.)

- a. *Modellierung einer Druckerwarteschlange*

Gehen Sie für die folgenden beiden Teilaufgaben von folgender Situation aus:

Ein Computer ist mit einem Drucker verbunden. Um zu vermeiden, dass der Computer einen Druckauftrag an den Drucker sendet, während er gerade noch mit einem anderen Druckauftrag beschäftigt ist, schickt der Computer seine Druckaufträge an eine Warteschlange. Wenn die Warteschlange nicht leer ist, entfernt der Drucker einen Druckauftrag aus der Warteschlange und druckt diesen.

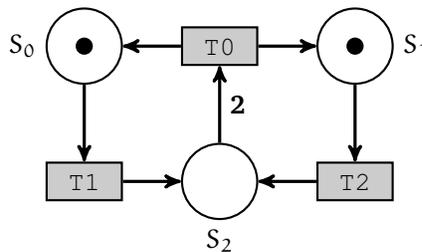
- (i) Modellieren Sie den oben beschriebenen Sachverhalt mit einem Petrinetz. Gehen Sie zudem von folgenden Bedingungen aus:
 - Der Computer kann immer nur einen Druckauftrag in die Warteschlange stellen. Er kann diesen Schritt aber beliebig oft wiederholen.
 - Der Drucker hat zwei Zustände: Entweder druckt er oder er wartet auf den nächsten Druckauftrag.
 - Die Warteschlange besitzt eine unendliche Kapazität, d.h. es können beliebig viele Druckaufträge in die Warteschlange gestellt werden.
 - Es spielt keine Rolle in welcher Reihenfolge der Drucker die Druckaufträge aus der Warteschlange entfernt.

Verwenden Sie zur Modellierung des Petrinetzes eine minimale Anzahl an Stellen, Marken und Transitionen.

- (ii) Damit die Warteschlange nicht überläuft, soll deren Kapazität nun begrenzt werden. Erweitern sie Ihr Petrinetz nun so, dass die Warteschlange nur noch maximal drei Aufträge akzeptiert. Verwenden Sie zur Modellierung wieder eine minimale Anzahl an Stellen, Marken und Transitionen. Führen Sie dabei **keine** Begrenzung der Kapazität der Stellen ein.

b. *Erstellung eines Erreichbarkeitsgraphen:*

Bearbeiten Sie die folgenden beiden Teilaufgaben in Bezug auf das unten abgebildete Petrinetz.



- (i) Geben Sie den Erreichbarkeitsgraphen für das dargestellte Petrinetz an. Verwenden Sie für eine Markierung M_j folgende Anordnung der Stellen $S_i (i \in \{0, \dots, 2\})$:

$$M_j = (S_0, S_1, S_2)$$

Der oben abgebildete Zustand entspricht also z.B. folgender Markierung:

$$M_0 = (1, 1, 0)$$

Kennzeichnen Sie jede Kante zwischen zwei Markierungen mit der entsprechenden Transition, die für den Übergang von einer Markierung zur anderen verantwortlich ist.

- (ii) Kann im dargestellten Petrinetz ein Deadlock entstehen? Begründen Sie Ihre Antwort.

Aufgabe 42: (H) Einfachauswahlaufgabe: Multiprocessing

(5 Pkt.)

Für jede der folgenden Fragen ist eine korrekte Antwort auszuwählen („1 aus n“). Eine korrekte Antwort ergibt jeweils einen Punkt. Mehrfache Antworten oder eine falsche Antwort werden mit 0 Punkten bewertet.

a) Was ist nach Coffman et al. [1] keine Voraussetzung für einen Deadlock?			
(i) Mutual Exclusion	(ii) Hold and Wait	(iii) No Preemption	(iv) Free Running
b) Für welche Anzahl an Prozessen n besteht die Gefahr eines Deadlocks, wenn jeder der n Prozesse von sechs vorhandenen Bandlaufwerken zwei beliebige Bandlaufwerke für seine Ausführung benötigt?			
(i) 3	(ii) 4	(iii) 5	(iv) 6
c) Was beschreibt die Deadlock-Situation beim 2 Philosophenproblem? Bei diesem kann Philosoph A bzw. B wahlweise denken oder essen. Auf dem Tisch werden 2 Stäbchen zur Verfügung gestellt, die ein Philosoph beide benötigt, um zu essen.			
(i) Philosoph A und B nehmen jeweils das (von ihnen aus) rechte Stäbchen auf	(ii) Philosoph A nimmt beide Stäbchen auf	(iii) Philosoph B nimmt beide Stäbchen auf	(iv) beide Stäbchen liegen auf dem Tisch

d) Welche allgemeine Aussage bezüglich der Kanten eines Petri-Netzes zur Prozessmodellierung ist korrekt?			
(i) Sie befinden sich zwischen je zwei Stellen.	(ii) Sie befinden sich zwischen je zwei Transitionen.	(iii) Eine Kante besteht zwischen genau einer Stelle s und einer Transition t , entweder von s nach t oder t nach s .	(iv) Die Kanten sind ungerichtet.
e) Wodurch wird die Dynamik eines Systems im Bezug auf Petri-Netze zur Prozessmodellierung beschrieben?			
(i) durch das Schalten von Stellen	(ii) durch das Schalten von Transitionen	(iii) durch das Schalten von Kapazitäten	(iv) durch das Schalten von Kantengewichten

Literatur

- [1] Edward G Coffman, Melanie Elphick, and Arie Shoshani. System deadlocks. ACM Computing Surveys (CSUR), 3(2):67–78, 1971.