



7. Übung zu Softwaretechnik 2

Management, Model Checking

1 Management

Beantworten Sie die folgenden Fragen zum Management der Softwaretechnik.

1.1 Unternehmensstruktur festlegen

Ein Softwarehaus erstellt Software fast ausschließlich für zwei Großkunden, die dem Softwarehaus jeweils die einzusetzenden Methoden und CASE-Systeme vorschreiben. Die Methoden und CASE-Systeme der Großkunden sind unterschiedlich. Welche Organisationsform wählen Sie, wenn Sie sicherstellen wollen, daß die Spezialisten ihr Wissen regelmäßig austauschen? (Betrachten Sie dazu die Gruppierungsalternativen!)

1.2 Ausbildung und Organisation

- Welche Auswirkungen auf eine Organisationsstruktur lassen sich im allgemeinen feststellen, wenn die Entscheidungsmöglichkeiten der Mitarbeiter beschnitten werden?
- Wann wird eine Arbeit als professionell bezeichnet? Welche Konsequenzen ergeben sich für die Ausbildung der Mitarbeiter, wenn sie eine professionelle Arbeit ausführen müssen?
- Welche Folgen hat eine starke horizontale Spezialisierung auf die Gestaltung von Positionen? Welche weiteren Arten der Spezialisierung kennen Sie? Von welchen Faktoren wird die Art der Spezialisierung beeinflusst?

2 Temporale Logik

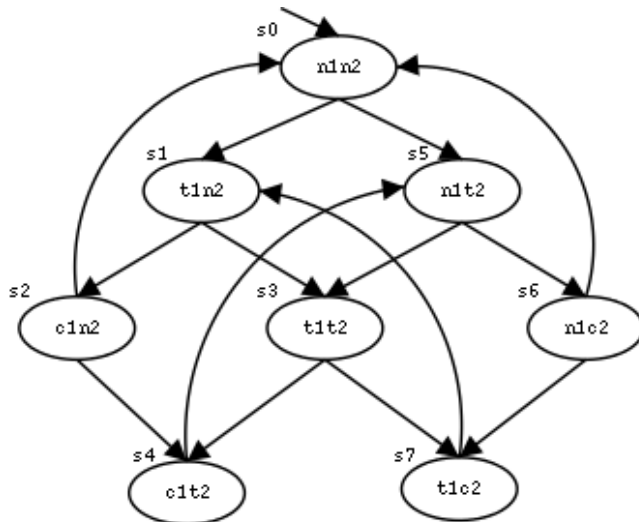
Formen Sie die Formeln in eine Form um, die nur aus **AF**, **EU** und **EX** sowie \wedge und \neg besteht. Die Formeln sind Teil der in der Vorlesung vorgestellten Verifikation des Corba-Systems.

- $\text{AG}(SRequest \rightarrow \text{AF}SReply)$
- $\text{AF}SReply \rightarrow \text{A}(\neg SReply \text{U}(SRequest \wedge \neg SReply))$

3 Model Checking

Beim konkurrierenden Zugriff auf exklusive Ressourcen hat ein Prozess 3 Zustände: *non-critical* (n), *trying* (t) und *critical* (c). Der Prozess arbeitet entweder an einer Aufgabe, die nicht auf die exklusive Ressource angewiesen ist (n), meldet sich für einen Zugriff auf die kritische Ressource an (t) oder arbeitet auf eben dieser Ressource, wobei zu diesem Zeitpunkt kein anderer Prozess darauf arbeiten darf (c).

In der Abbildung findet sich ein Modell für den exklusiven Zugriff auf eine kritische Ressource von 2 Prozessen. Das in der Abbildung dargestellte Modell soll folgende Eigenschaften erfüllen:



Sicherheit (Safety) Es greift immer nur ein Prozess auf die Ressource zu.

Lebendigkeit (Liveness) Wenn ein Prozess eine Ressource anfordert, kriegt er sie auch irgendwann.

Nicht blockierend (Non-blocking) Ein Prozess kann jederzeit eine Ressource anfordern.

beliebige Reihenfolge (No strict sequencing) Die Reihenfolge der Prozesse für den Zugriff auf die Ressource ist nicht festgelegt.

Aufgabe

- Stellen Sie für jede Eigenschaft eine temporallogische Formel für den ersten Prozess auf.
- Entfalten Sie den Automaten in einen Berechnungsbaum und stellen Sie verschiedene Pfade auf. Überlegen Sie sich geeignete Abkürzungen und Darstellungen für Schleifen.
- Beweisen Sie mit dem in der Vorlesung vorgestellten naiven Algorithmus zum Model-Checking, dass die Lebendigkeit nicht gilt.
- Verbessern Sie das Modell um die Lebendigkeit zu garantieren.

4 Kreativität *

Modellieren Sie die Eigenschaften einer Tankstellen-Zapfsäule (z.B. mit den Elementen Pumpe, Zapfpistole, Aufhängung der Zapfpistole, Kasse). Beweisen Sie, dass es in Ihrem Modell nicht möglich ist, dass ein anderer Kunde tankt, bevor der erste Kunde bezahlt hat.

Fragen?

Bei Fragen wenden sie sich an Holger Cleve <cleve@cs.uni-sb.de>.