4. Aufgabenblatt zur Vorlesung

Nichtsequentielle und Verteilte Programmierung

SoSe 2018

Wolfgang Mulzer, Katinka Wolter

Abgabe am 18. Mai 2018 bis 12 Uhr in die Tutorenfächer

Aufgabe 1 Lineare Temporale Logik I *10 Punkte*

(a) Gilt die folgende Formel für alle möglichen Belegungen? Begründen Sie Ihre Antwort.

$$\Diamond\Box\, p \Leftrightarrow \Box\Diamond\, p$$

(b) Gilt die folgende Formel für alle möglichen Belegungen? Begründen Sie Ihre Antwort.

$$(\Box\, p \Rightarrow \Diamond\, q) \wedge \Diamond\Box\, q \Rightarrow \Diamond\, q$$

(c) Beweisen Sie, dass die folgenden Formeln für alle möglichen Belegungen gelten.

(i) $(\Box\, p \wedge \Box\, q) \Leftrightarrow \Box(p \wedge q)$

(ii) $(\Diamond\, p \vee \Diamond\, q) \Leftrightarrow \Diamond(p \vee q)$

Aufgabe 2 Lineare Temporale Logik II *10 Punkte*

In der linearen temporalen Logik gibt es oft noch die zusätzlichen Operatoren *bis* ($\mathcal{U}$) und *nächster* ($\bigcirc$).

Die informelle Spezifikation ist wie folgt: Seien α und β zwei LTL Formeln. Dann ist $\alpha\,\mathcal{U}\,\beta$ genau dann wahr zu einem Zeitpunkt i, $i \in \mathbb{N}$, wenn ein Zeitpunkt $j \geq i$ existiert, so dass β zum Zeitpunkt j und α für alle Zeitpunkte $k = i, \ldots, j-1$ wahr ist.

Es ist $\bigcirc\,\alpha$ genau dann wahr zu einem Zeitpunkt i, $i \in \mathbb{N}$, falls α zum Zeitpunkt $i+1$ wahr ist.

(a) Erweitern Sie die formale Syntax und Semantik der linearen temporalen Logik aus der Vorlesung um die Operatoren $\mathcal{U}$ und $\bigcirc$. Geben Sie zwei Beispiele.

(b) Warum eignet sich der Operator $\bigcirc$ nicht zur Spezifikation von Fortschrittseigenschaften? Was ist das Problem mit einer Formel der Form $\Box\big[a_1 \Rightarrow \bigcirc\, a_2\big]$?

(c) Definieren Sie $\Box$ und $\Diamond$ mit Hilfe von $\mathcal{U}$ (und den elementaren Operatoren der klassischen Aussagenlogik) und beweisen Sie die Korrektheit ihrer Definition.

Aufgabe 3 Der Algorithmus von Peterson II *10 Punkte*

Der Algorithmus von Peterson (Aufgabe 3.2) lautete folgendermaßen:

```
=====================================================
  global adrin = false, bdrin = false, letzter = a
a1: U                       | b1: U
a2: adrin  <- true          | b2: bdrin  <- true
a3: letzter <- a            | b3: letzter <- b
a4: while bdrin &&          | b4: while adrin &&
          letzter = a do    |           letzter = b do
a5:     NOP                 | b5:     NOP
a6: K                       | b6: K
a7: adrin <- false          | b7: bdrin <- false
```

(a) Beweisen Sie, dass im Algorithmus von Peterson die folgenden Invarianten gelten:

- $(a_6 \wedge (b_4 \vee b_5)) \Rightarrow (\texttt{adrin} \wedge \texttt{letzter} = b)$
- $((a_4 \vee a_5) \wedge b_6) \Rightarrow (\texttt{bdrin} \wedge \texttt{letzter} = a)$

(b) Argumentieren Sie, dass die folgenden Formeln für jede mögliche Zustandsfolge gelten:

- $\Box\big[(a_4 \wedge \Box\neg a_6) \Rightarrow \Box\Diamond(\texttt{bdrin} \wedge \texttt{letzter} \neq b)\big]$,
- $\Box\big[\Diamond\Box(\neg\texttt{bdrin}) \vee \Diamond(\texttt{letzter} = b)\big]$, und
- $\Box\big[(a_4 \wedge \Box\neg a_6 \wedge \Diamond(\texttt{letzter} = b)) \Rightarrow \Diamond\Box(\texttt{letzter} = b))\big]$.

Sie können ggf. die Invarianten aus (a) verwenden.

(c) (*freiwillig, 5 Zusatzpunkte*) Beweisen Sie, dass Petersons Algorithmus korrekt ist.