Lösungsvorschlag Aufgabe 3: Ablaufpläne

R-Ablaufplan :

Die Menge $bl_R(\alpha)$ ist die Menge der Aktionen, d.h. deren Ausführung vergeblich versucht wurde und die danach im restlichen Ablauf nicht ausgeführt wurden:

$$bl_R(\lambda) = \varnothing$$

$$bl_R(\alpha \cdot a) = \begin{cases} bl_R(\alpha) \cup \{a\} & \text{falls } a \notin lf_P(\alpha) \\ bl_R(\alpha) \setminus \{a\} & \text{falls } a \in lf_P(\alpha) \end{cases}$$

Der R-Ablaufplan läßt nun alle Abläufe α zu, die folgender Bedingung genügen:

$$\alpha \text{ semiaktiv } \wedge \ \forall\, i \in \mathbb{N}\ (1 \le i \le \text{Länge}(\ \alpha\ ) \wedge\ bl_R({}_{i-1}\alpha)\ \cap\ lf_P({}_{i-1}\alpha) \neq \varnothing$$
$$\Rightarrow \alpha_i \in\ bl_R({}_{i-1}\alpha)\ \cap\ lf_P({}_{i-1}\alpha)\ \ )$$

a) Der Ablauf $\alpha = (a_{1,1} \bullet a_{2,1} \bullet a_{3,1} \bullet a_{4,1} \bullet a_{1,2} \bullet a_{2,1} \bullet a_{3,1} \bullet a_{4,1})$:

	S_1	S_2	S_3	lf_P	bef_P	bl_R	$lf_P \cap bl_R$
I	1	1	1	$\{a_{1,1},a_{2,1},a_{3,1},a_{4,1}\}$	$\{\mathbf{a_{1,1}},a_{2,1},a_{3,1},a_{4,1}\}$	$\varnothing$	$\varnothing$
$a_{1,1}$	1	0	0	$\{a_{1,2}\}$	$\{a_{1,2},\mathbf{a_{2,1}},a_{3,1},a_{4,1}\}$	$\varnothing$	$\varnothing$
$a_{2,1}$	1	0	0	$\{a_{1,2}\}$	$\{a_{1,2},a_{2,1},\mathbf{a_{3,1}},a_{4,1}\}$	$\{a_{2,1}\}$	$\varnothing$
$a_{3,1}$	1	0	0	$\{a_{1,2}\}$	$\{a_{1,2},a_{2,1},a_{3,1},\mathbf{a_{4,1}}\}$	$\{a_{2,1},a_{3,1}\}$	$\varnothing$
$a_{4,1}$	1	0	0	$\{a_{1,2}\}$	$\{\mathbf{a_{1,2}},a_{2,1},a_{3,1},a_{4,1}\}$	$\{a_{2,1},a_{3,1},a_{4,1}\}$	$\varnothing$
$a_{1,2}$	1	1	1	$\{a_{1,3},a_{2,1},a_{3,1},a_{4,1}\}$	$\{a_{1,3},\mathbf{a_{2,1}},a_{3,1},a_{4,1}\}$	$\{a_{2,1},a_{3,1},a_{4,1}\}$	$\{\mathbf{a_{2,1}},a_{3,1},a_{4,1}\}$
$a_{2,1}$	0	1	0	$\{a_{1,3},a_{2,2},a_{4,1}\}$	$\{a_{1,3},a_{2,2},a_{3,1},\mathbf{a_{4,1}}\}$	$\{a_{3,1},a_{4,1}\}$	$\{\mathbf{a_{4,1}}\}$
$a_{3,1}$							nicht möglich
$a_{4,1}$							

α ist unter einem R-Ablaufplan nicht ablauffähig.

b) Der Ablauf $\alpha = (a_{1,1} \bullet a_{1,2} \bullet a_{2,1} \bullet a_{3,1} \bullet a_{1,3} \bullet a_{2,1} \bullet a_{2,2})$:

	S	lf_P	bef_P	bl_R	$lf_P \cap bl_R$
I	1	$\{a_{1,1},a_{2,1},a_{3,1}\}$	$\{\mathbf{a_{1,1}},a_{2,1},a_{3,1}\}$	$\varnothing$	$\varnothing$
$a_{1,1}$	0	$\{a_{1,2}\}$	$\{\mathbf{a_{1,2}},a_{2,1},a_{3,1}\}$	$\varnothing$	$\varnothing$
$a_{1,2}$	0	$\{a_{1,3}\}$	$\{a_{1,3},\mathbf{a_{2,1}},a_{3,1}\}$	$\varnothing$	$\varnothing$
$a_{2,1}$	0	$\{a_{1,3}\}$	$\{a_{1,3},a_{2,1},\mathbf{a_{3,1}}\}$	$\{a_{2,1}\}$	$\varnothing$
$a_{3,1}$	0	$\{a_{1,3}\}$	$\{\mathbf{a_{1,3}},a_{2,1},a_{3,1}\}$	$\{a_{2,1},a_{3,1}\}$	$\varnothing$
$a_{1,3}$	1	$\{a_{1,4},a_{2,1},a_{3,1}\}$	$\{a_{1,4},\mathbf{a_{2,1}},a_{3,1}\}$	$\{a_{2,1},a_{3,1}\}$	$\{\mathbf{a_{2,1}},a_{3,1}\}$
$a_{2,1}$	0	$\{a_{1,4},a_{2,2}\}$	$\{a_{1,4},\mathbf{a_{2,2}},a_{3,1}\}$	$\{a_{3,1}\}$	$\varnothing$
$a_{2,2}$	0	$\{a_{1,4},a_{2,3}\}$	$\{a_{1,4},a_{2,3},a_{3,1}\}$	$\{a_{3,1}\}$	$\varnothing$

nur im Fall $bl_R\ ({}_{i-1}\alpha\ )\ \cap lf_P\ ({}_{\iota-1}\alpha\ ) \neq \varnothing$ verlangt der R-Ablaufplan die nächste Aktion aus der Schnittmenge. Diese Vorgehensweise wird durch den Ablauf α realisiert, wodurch er unter einem R-Ablaufplan ablauffähig ist.

Lösungsvorschlag Aufgabe 4:

a) Ordnen Sie den folgenden informellen Aussagen eine der in der Vorlesung vorgestellten Implementierungseigenschaften zu!

<u>In der in der Frage angegebenen Reihenfolge:</u>
(mit einer umgangsprachlichen Beschreibung der Definition oder eines beschreibenden Satzes der jeweiligen Eigenschaft)

sicher

Wenn die Implementierung in einem bestimmten Zustand mit einer bestimmten, beobachtbaren Aktion fortsetzbar ist, dann ist auch die Spezifikation im entsprechenden Zustand mit dem Abbild der beobachtbaren Aktion fortsetzbar (Satz 2.4).

vollständig

Die Menge der beobachtbaren Abläufe unter einem Ablaufplan $\overline{S}$ in der Implementierung abgebildet auf die Spezifikation muß eine Obermenge der in der Spezifikation möglichen Abläufe unter dem eventuell verschiedenen Ablaufplan S sein.

verklemmungsfrei
Falls ein aktiver Ablauf in der Implementierung nicht fortsetzbar ist, so ist sein Abbild auch nicht in der Spezifikation fortsetzbar.

verklemmungskonsistent
Wenn zu einem aktiven Ablauf in der Implementierung nur noch Fortsetzungen gefunden werden können, die nicht beobachtbar sind, dann muß die Spezifikation verklemmt sein (Anmerkung: das ist mehr als verklemmungsfrei).
Verklemmt sich die Spezifikation, so muß sich die Implementierung nach endlich vielen Buchhaltungsaktionen auch verklemmen.

prompt
In aktiven Abläufen ist die Zahl der nicht beobachtbaren Aktionen, die zwischen zwei beobachtbaren Aktionen ausgeführt werden, durch eine obere Schranke beschränkt.

behinderungsfrei
Wenn ein Teilprozeß nach einem bestimmten Ablauf in der Implementierung nicht lauffähig ist, so ist der entsprechende Teilprozeß in der Spezifikation nach dem dazugehörigen, abgebildeten Ablauf in der Spezifikation auch nicht lauffähig.

prozeßtreu
Genau dann wenn zwei beobachtbare Aktionen in der Implementierung zu einem Teilprozeß gehören, dann müssen ihre Abbilder in der Spezifikation auch zu einem Teilprozeß gehören.

schwach prozeßtreu
Wenn zwei beobachtbare Aktionen in der Implementierung zu einem Teilprozeß gehören, dann müssen ihre Abbilder in der Spezifikation auch zu einem Teilprozeß gehören.

b) Welche Implementierungseigenschaften sollte ein Prozeßsystem *auf jeden Fall* haben?

sicher
Wenn in der Implementierung beliebige, nicht spezifizierte Abläufe auftreten, dann können anhand der Spezifikation keine Aussagen gemacht werden und die Implementierung ist wertlos.

verklemmungsfrei
Die Implementierung sollte keine Programmierfehler enthalten, die zu Verklemmungen führen.

prompt
Durch *busy wait* wird sinnlos Rechenzeit verschwendet, die für andere Arbeiten genutzt werden könnte.

c) Welche Implementierungseigenschaften bei einem Prozeßsystem sind *wünschenswert*?

vollständig
Auf höheren Abstraktionsebenen ist es sinnvoll, vollständige Implementierungen zu fordern, da diese den größtmöglichen Freiraum bei der Implementierung auf den unteren Ebenen lassen.

verklemmungskonsistent
Diese Eigenschaft ist sehr wünschenswert, aber nur wenn man die Möglichkeit hat, die Spezifikation zu ändern und dort vorhandene Verklemmungen zu entfernen.

behinderungsfrei
Die in der Spezifikation mögliche Parallelarbeit ist auch in der Implementierung möglich und wird z.B. nicht durch zusätzliche kritische Abschnitte behindert.
Im Zusammenhang mit prompt ist es möglich, Zeitabschätzungen anhand der Spezifikation abzuleiten.

prozeßtreu
Wenn in der Spezifikation von einem Prozeßkonzept Gebrauch gemacht wird, um etwa die Last auf mehrere Prozessoren zu verteilen oder Synchronisationsmaßnahmen durch eine Hintereinanderausführung kritischer Abschnitte einzusparen, so ist Prozeßtreue zu fordern. Werden Prozesse nur dazu benutzt, um ein Problem realitätsnäher zu spezifizieren, so kann eventuell auf Prozeßtreue verzichtet werden.

d) Welche Implementierungseigenschaften können in bestimmten Situationen nicht erfüllt werden?

vollständig
Implementierungen von Prozeßsystemen auf Monoprozessoren können im Allgemeinen nicht vollständig sein, da sie nur einen einzigen aktiven Ablauf zulassen (Deterministischer Scheduler).

behinderungsfrei
Nur mit PV_1-Operationen sind behinderungsfreie Lösungen im Allgemeinen nicht möglich.

prompt
Auf Multiprozessoren werden i.A. statt blockierenden PV-Operationen Spin Locks verwendet.

Was gibt es für Abläufe?

Ein **Ablauf** zu einem Prozeßsystem P ist definiert als eine beliebige Folge von Aktionen aus A. Bei einem **aktiven Ablau**f ist jede Aktion an der Stelle, wo sie steht, ausführbar. Bei einem **semiaktiven Ablauf** steht sie zumindest zur Ausführung an. Für den Rechnerbetrieb sind ausschließlich die zumindest semiaktiven Abläufe interessant, da andere Abläufe bei (sequentiellen) Prozessen nicht vorkommen.

Was ist ein Ablaufplan ?

Durch den **Ablaufplan**, den ein Betriebssystem verwendet, wird die Menge der erlaubten Abläufe eingeschränkt.
Ein Ablaufplan ist definiert als ein Prädikat über der Menge der Abläufe. Praktisch stellt er eine Strategie dar, mit der das Betriebssystem nach jeder ausgeführten Aktion ermittelt, welche der anstehenden Aktionen als nächste ausprobiert werden soll. Ein "richtiger" Ablaufplan ist also ein Prädikat über der Menge der semiaktiven bzw. aktiven Abläufe. Die semiaktiven Ablaufpläne spielen eine größere Rolle in realen Betriebssystemen, denn man kann bei einem Monoprozessor im allgemeinen nicht vorher wissen, ob die NBL einer Aktion erfüllt ist oder nicht.

Zur Veranschaulichung: