

Fragen zur Systemprogrammierung

Teil C – Vorlesung 10.2

Prozesssynchronisation / Monitore

1. Welches Wissen bringe ich aus den vorangegangenen Vorlesungen mit?¹
2. Welche grundlegenden Eigenschaften besitzt ein Monitor, wie kommt mehrseitige Synchronisation ins Spiel und inwiefern findet einseitige Synchronisation statt? Welche dieser Synchronisationsarten ist hier zwingend, welche optional? (S. 7)
3. Inwiefern ist ein Monitor einer Klasse beziehungsweise einem Modul ähnlich? (S. 8)
4. Wie sind Monitorprozeduren repräsentiert, welche Rolle spielt dabei ein Kompilierer und welchen Gültigkeitsbereich haben die expliziten Synchronisationsanweisungen? (S. 9)
5. Welche Sorten von Warteschlangen sind eng mit dem Monitorkonzept verknüpft, welche Queue verzeichnet welche Prozessexemplare, warum muss ein solches darauf verzeichnetes Exemplar außerhalb des Monitors warten? (S. 10)
6. Welche Gemeinsamkeiten und Unterschiede haben Monitore nach Hansen und Hoare? (S. 11)
7. Wie funktioniert ein Mesa-Monitor? (S. 12)
8. Inwiefern begründet die Semantik einer Bedingungsvariable Monitorunterschiede, wie wirkt die Signalisierung bei Hansen-, Hoare- und Mesa-Monitoren? (S. 13)
9. Wie unterscheidet sich ein Java-Monitor von Hansen-, Hoare- und Mesa-Monitoren? (S. 14)
10. Wozu dient eine Bedingungsvariable, wie funktionieren die auf sie typischerweise definierten Operationen und welcher Unterschied besteht dabei zur Ereignisvariablen? (S. 16)
11. Unter welcher Voraussetzung ist nach Auflösung der Wartebedingung diese wiederholt zu überprüfen und weshalb muss ein Prozess, dessen Wartebedingung erfüllt ist, den Monitor freigegeben haben, bevor ein anderer Prozess den Prozessor erhält? (S. 17)
12. Wie wird die Aufhebung einer Wartebedingung innerhalb eines Monitors bewirkt, was geschieht, wenn für keinen Prozess diese Wartebedingung zutrifft und was, wenn sie für mehr als einen Prozess gilt? (S. 18)
13. Was bedeutet der Besitzwechsel eines Monitors, wann geschieht dieser und welche Art von Monitor führt diesen in welcher Weise durch? (S. 19)
14. Was bedeutet demgegenüber die Besitzwahrung eines Monitors, wann erhält ein so signalisierter Prozess frühestens den Monitorzutritt und unter welcher Voraussetzung muss er seine Wartebedingung nicht erneut überprüfen? (S. 20)
15. Weshalb beugt ein als Monitor implementierter „*bounded buffer*“ dem „*lost wake-up*“-Problem vor und in welchem Kontext innerhalb einer Monitorprozedur tritt diese potentielle Fragestellung überhaupt zutage? (S. 22)
16. Welcher Stoff wurde in dieser Vorlesung vermittelt, was sollte ich gelernt haben?¹

¹Diese Frage ist nicht prüfungsrelevant.