

1 Thema: Prozesse und Threads

1.1 Konzept der Prozesse und Threads (4 Punkte)

Beschreiben Sie die Unterschiede zwischen Kernel-Level-Threads und User-Level-Threads.
[Schreiben Sie maximal eine halbe bis eine ganze Seite]

1.2 Sinnvoller Einsatz von Threads (2 Punkte)

Beschreiben Sie zwei sinnvolle Beispiele für den Einsatz von Threads.
[Schreiben Sie maximal eine halbe Seite]

2 Thema: Gegenseitiger Ausschluss (18 Punkte)

Algorithmus von Peterson

Auf der Materialiensseite zur Vorlesung finden Sie den Algorithmus von Peterson, wie er in der Vorlesung eingeführt wurde, ausprogrammiert für pthreads. Die Prozesse A und B, hier realisiert durch pthreads, konkurrieren um eine Ressource und benutzen sie regelmäßig. Wie man sich an der Ausgabe leicht überzeugen kann, gibt es weder eine Verklemmung noch ein Aushungern eines einzelnen Threads. Sie können das ganze etwas interessanter im Ablauf gestalten, indem Sie z.B. bei Prozess A in den kritischen Abschnitt der Schleife (gekennzeichnet durch den Kommentar *critical section*) die Anweisung `sleep(1);` einfügen. Dann kommt B viel häufiger in den kritischen Abschnitt.

Ihre Aufgabe: Erweitern Sie das Programm auf 4 Threads A, B, C und D, so dass weder Verklemmung noch Aushungern auftritt. Erweitern sie den Peterson Algorithmus hierbei auf 4 Threads und verwenden Sie keine vom Betriebssystem zur Verfügung gestellten Mechanismen zur Synchronisation (wie z.b. Mutex).

Hinweise:

- Der Quellcode `pthread-2-peterson.c` kann von der Materialiensseite der Vorlesung heruntergeladen werden.
- Das Programm kann so kompiliert werden: `gcc -o <NameOfBinary> pthread-2-peterson.c -lpthread`
- Sollte ihnen kein Linux System zur Verfügung stehen, besorgen Sie sich KNOPPIX. Dabei handelt es sich um eine freie GNU/Linux-Distribution, die sich als Live-CD direkt von einer CD/DVD starten lässt. Eine Installation auf der Festplatte ist nicht notwendig: <http://www.knopper.net/knoppix>.