



Systemsoftware I Übungsblatt 3

POSIX.4-Scheduling

1. Entwerfen Sie ein Programm `aLoad`, das periodisch jeweils einen CPU- und einen I/O-Burst vorgegebener Länge aufweist. Die Burstlängen sollen als zwei Programmargumente in der Auflösung von Mikrosekunden übergeben werden. Ein drittes Argument gibt die Gesamtlaufzeit des Programms in Sekunden vor.

Das Programm soll die Bursts überwachen und den maximalen Fehler alle 10 Sekunden in einer geeigneten Datenstruktur im Speicher festhalten. Mit Ablauf der Gesamtlaufzeit sollen die ermittelten Fehler ausgegeben werden.

- a) Ermitteln Sie die maximal erreichbare Auflösung für einen CPU-Burst.
- b) Ermitteln Sie die maximal erreichbare Auflösung für einen I/O-Burst.

2. Starten Sie einen `aLoad`-Prozeß mit einem beliebigen CPU-Burst, und einem I/O-Burst der Länge 0.

- a) Verwenden Sie die höchste Priorität in der FIFO-Klasse.
- b) Verwenden Sie die höchste Priorität in der RR-Klasse.
- c) Verwenden Sie die normale Priorität in der OTHER-Klasse.

Ermitteln Sie jeweils die maximalen Burstfehler (1) wenn nur der Prozeß `aLoad` auf dem System aktiv ist und (2) wenn parallel eine Instanz des Speicherblockierers aus Übung 1 ausgeführt wird.

3. Ermitteln Sie die maximalen Burstfehler bei folgendem Lastprofil

- `aLoad` mit einem CPU-Burst von 300 ms und einem I/O-Burst von 700 ms.
- `aLoad` mit einem CPU-Burst von 900 ms und einem I/O-Burst von 1100 ms.

- a) Ordnen Sie beiden `aLoad`-Instanzen geeignete Prioritäten in der FIFO-Klasse zu.
- b) Verwenden Sie geeignete Prioritäten in der RR-Klasse.
- c) Verwenden Sie für beide Prozesse die Standardprioritäten der OTHER-Klasse.

Führen Sie die Messungen (1) nur mit den beiden `aLoad`-Prozessen und (2) parallel mit einem Speicherblockierer aus Übung 1 durch.