

Hochschule RheinMain
Fachbereich DCSM - Informatik
Prof. Dr. Robert Kaiser
Sebastian Flothow
Alexander Schönborn
Daniel Schultz

Betriebssysteme

WS 2020/21

LV 3122

Übungsblatt 1

Bearbeitungszeit 1 Woche

Keine Abgabe, Besprechung der Aufgaben in der Praktikumsveranstaltung

Hinweis: Die Aufgaben dieses Übungsblatts beziehen sich auf die Kapitel 1 bis 5 der Vorlesung.

Aufgabe 1.1 (Betriebsarten):

- a) Welche Betriebsarten werden in der Entwicklung von Rechensystemen unterschieden?
- b) Erläutern Sie die prinzipielle Arbeitsweise des Mehrprogrammbetriebs (Multiprogramming). Nennen Sie Gründe für die Einführung.

Aufgabe 1.2 (Benutzer- und Kernmodus):

- a) Was bedeuten die Begriffe Benutzermodus und Kernmodus?
- b) Welche der folgenden Operationen sollten nur im Kernmodus erlaubt sein:
 - b1) Maskieren aller Unterbrechungen
 - b2) Lesen der Tageszeituhr
 - b3) Betriebssystemkernaufwurf
 - b4) Stellen der Tageszeituhr
 - b5) Verändern der Speicherfunktion zur Abbildung virtueller Adressen auf reale (i.d.R. als Seitentabellen realisiert).
- c) Erläutern Sie den Ablauf des Eintritts in den Betriebssystemkern aus einem C-Anwendungsprogramm heraus bei Aufruf eines system calls (gekapselt durch die entsprechende C-Bibliotheksfunktion).

Aufgabe 1.3 (Betriebssystemansichten):

Die in der Vorlesung gegebene Definition eines Betriebssystems beinhaltet zwei Ansichten auf ein Betriebssystem. Welche sind es, und was bedeuten sie?

Aufgabe 1.4 (Betriebssystemstrukturen):

Welche internen Strukturierungsprinzipien von Betriebssystemen kennen Sie? Was sind die charakteristischen Merkmale, Vor- und Nachteile? Nennen Sie Beispiele.

Aufgabe 1.5 (UNIX-Kern):

- a) Warum kann man den UNIX-Kern als prozedurorientiert bezeichnen?
- b) Warum ist der UNIX-Kommando-Interpreter (shell) nicht Teil des Betriebssystemkerns, wohl aber des Betriebssystems?

Aufgabe 1.6 (Einsatzspektrum heutiger Betriebssysteme):

Erläutern Sie das Einsatzspektrum heutiger Betriebssysteme.

Aufgabe 1.7 (Prozesse):

- a) Wie werden Prozesse innerhalb des UNIX-Betriebssystems repräsentiert, und wie werden sie identifiziert?
- b) Was ist der Unterschied zwischen Prozessen und Threads?
- c) Was ist der Unterschied zwischen Traps und Interrupts? Was sind Gemeinsamkeiten?
- d) Warum kann es sinnvoll sein, die Priorität eines Prozesses dynamisch festzulegen? Ist das immer der Fall, oder gibt es auch Anwendungen, für die das nicht sinnvoll ist?

Aufgabe 1.8 (Scheduling):

Die folgenden Aufträge treffen in einem Einprozessorsystem nahezu zur gleichen Zeit in der angegebenen Reihenfolge ein. Eine höhere Zahl für die Priorität drücke eine höhere Wichtigkeit aus. Die Aufträge seien reine Rechenaufträge ohne I/O. Die Prozesswechselzeiten werden vernachlässigt.

| Auftrag | Bedienzeit | Priorität |
|---------|------------|-----------|
| 1 | 10 | 3 |
| 2 | 6 | 5 |
| 3 | 2 | 2 |
| 4 | 4 | 1 |
| 5 | 8 | 4 |

Geben Sie für die Non-Preemptiven-Scheduling-Verfahren

- a) FCFS,
- b) Prioritäten,

- c) Shortest-Job-First sowie für
- d) Round-Robin mit Quantum $Q = 3$

ein (CPU-Belegungs)-Diagramm für die Ausführung der Prozesse an, bestimmen Sie die jeweilige Verweilzeit sowie die mittlere Wartezeit.

Aufgabe 1.9 (Wechselseitiger Ausschluss):

- a) Was beinhaltet der Begriff „wechselseitiger Ausschluss“?
- b) Was ist ein kritischer Abschnitt?
- c) Erläutern Sie die Eigenschaften eines „guten“ Algorithmus zum wechselseitigen Ausschluss. Warum macht es Sinn, neben der funktionalen Korrektheit weitere Eigenschaften zu fordern?
- d) Erklären Sie den Unterschied zwischen Aktivem Warten und Blockieren.

Aufgabe 1.10 (Semaphore):

Skizzieren Sie in Pseudo-Code, wie in einem Betriebssystem Semaphore implementiert werden können, wenn als Basis das Maskieren aller Unterbrechungen zur Durchsetzung ununterbrochener Befehlsfolgen dienen. Sie können dabei Kern-interne Funktionen benutzen, wie in Folie 5-27 beschrieben.