

Hochschule RheinMain
Fachbereich DCSM - Informatik
Prof. Dr. Robert Kaiser
Kai Beckmann
Tanjef Moos
Alexander Schönborn
Ingo Speer

Betriebssysteme

WS 2021/22

LV 3122

Übungsblatt 2

Bearbeitungszeit 2 Wochen

Keine Abgabe, Besprechung der Aufgaben in der Praktikumsveranstaltung

Hinweis: Die Aufgaben dieses Übungsblatts beziehen sich auf die Kapitel 5 bis 10 der Vorlesung.

Aufgabe 2.1 (Scheduling):

Die folgenden Aufträge treffen in einem Einprozessorsystem zu den angegebenen Zeitpunkten ein. Eine höhere Zahl für die Priorität drücke eine höhere Wichtigkeit aus. Die Aufträge seien reine Rechenaufträge ohne I/O-Anforderungen. Die Prozesswechselzeiten werden vernachlässigt.

Auftrag	Ankunftszeit	Bedienzeit	Priorität
1	0	5	3
2	3	2	5
3	6	2	4
4	10	4	1
5	12	3	4
6	13	1	2

Geben Sie für das Preemptive Prioritäts-Scheduling-Verfahren ein (CPU-Belegungs)-Diagramm für die Abarbeitung der Prozesse an und bestimmen Sie die mittlere Verweilzeit.

Aufgabe 2.2 (Scheduling):

Bearbeiten Sie Aufgabe Aufgabe 2.1 für den Fall eines 2-Prozessorsystems, wenn jeder der beiden Prozessoren die halbe Rechenleistung gegenüber dem Prozessor aus Aufgabe Aufgabe 2.1 besitzt. Wie verändert sich die mittlere Verweilzeit?

Aufgabe 2.3 (Scheduling):

Die folgenden Aufträge treffen in einem Einprozessorsystem zu den angegebenen Zeitpunkten ein. Die Aufträge seien reine Rechenaufträge ohne I/O-Anforderungen. Die Prozesswechselzeiten werden vernachlässigt.

Auftrag	Ankunftszeit	Bedienzeit
1	0	6
2	3	2
3	6	6
4	7	3
5	12	4
6	13	2

Unter SRJF (Shortest Remaining Job First) werde die unterbrechende Variante von SJF verstanden. Geben Sie für SRJF ein (CPU-Belegungs)-Diagramm für die Abarbeitung der Prozesse an, und bestimmen Sie die mittlere Wartezeit und die mittlere Verweilzeit.

Aufgabe 2.4 (Prozesskommunikation):

Ein Betriebssystem stellt ein Nachrichtensystem mit den Primitiven `send()` und `receive()` mit direkter asymmetrischer Adressierung der Kommunikationspartner bereit (vgl. Vorlesung), bietet aber bisher keine Semaphoren. Skizzieren Sie eine Realisierung von Semaphoren mit den Operationen $P()$ und $V()$, die auf Nachrichtenaustausch mit einem „Semaphor-Server-Prozess“ SSP basiert, der alle zur Verfügung gestellten „Semaphoren“ realisiert. Geben Sie dessen Algorithmus ebenfalls an.

Aufgabe 2.5 (Prozesskommunikation):

Ausgangspunkt ist das Erzeuger-Verbraucher-Problem (vgl. Vorlesung). Betrachten Sie die Variante, die anstelle des beschränkten Puffers einen Puffer mit unbeschränkter Kapazität annimmt. Geben Sie eine Lösung mit Semaphoren für die Algorithmen von Erzeuger und Verbraucher an. Modifizieren Sie dazu die gegebene Lösung des Ausgangsproblems.

Aufgabe 2.6 (Prozesskommunikation):

Die in der Vorlesung angegebene Lösung des Leser-Schreiber-Problems bevorzugt Leser. Wie könnte die angegebene Semaphoren-Lösung erweitert werden, so dass ein Schreiber vor allen nach ihm ankommenden Lesern bedient wird, er also nicht von Lesern „überholt“ werden kann?

Aufgabe 2.7 (Deadlocks):

- Welche Formen des Umgangs mit dem Deadlock-Problem kennen Sie? Diskutieren Sie die konzeptionellen Unterschiede.
- Was beinhaltet die geordnete Betriebsmittelbenutzung? Was erreicht sie und wodurch?
- Warum wird die geordnete Betriebsmittelbenutzung in Betriebssystemkernen eingesetzt?

Aufgabe 2.8 (Deadlocks):

Gegeben sei der Betriebsmittelvektor $E = (5221)$. Die aktuelle Belegungsmatrix C und die Anforderungsmatrix R seien:

C:	(a) R:	(b) R:
0 1 1 0	2 0 1 1	2 0 1 1
2 0 0 1	1 0 1 0	1 2 1 0
0 1 1 0	2 0 0 0	2 0 0 0

Beantworten Sie für die Fälle (a) und (b): Ist der aktuelle Zustand ein Deadlock-Zustand? Wenn ja, welche Prozesse sind in den Deadlock verstrickt? Wenn nein, ist der Zustand sicher?

Aufgabe 2.9 (Caches):

- a) Wie groß ist die effektive Wartezeit eines Systems mit einem einstufigen Cache, wenn der Hauptspeicherzugriff 8 Prozessortakte Wartezeit erfordert und der Zugriff auf den Cache nur einen Wartetakt. Die Trefferrate des Caches sei 70%, die Taktfrequenz des Prozessors 200 MHz, d.h. ein Takt dauert $\frac{1}{200 \cdot 10^6 s^{-1}} = 5ns$.
- b) Im Folgenden werden Caches verschiedener Organisationsformen betrachtet. Für alle Beispiele gilt (vgl. Vorlesung):

- 32 Adressbits
- Cache-Größe: 2048 Byte
- Cache-Zeilengröße: 16 Byte

Bei einem vollassoziativen, direkt abbildenden oder zweifach assoziativen Cache werde jeweils auf ein 32-Bit-Wort unter der Speicheradresse 0x002A2564 zugegriffen. Für alle drei Fälle soll beantwortet werden:

- b1) Auf welche Cache-Zeile wird zugegriffen?
- b2) Wie lautet der beim Vergleich benutzte Tag
- b3) Wo in der Cache-Zeile steht das adressierte 32-Bit-Wort?
- c) Der direkt abbildende Cache aus der vorigen Teilaufgabe werde als L1 Datencache benutzt. Wie groß sind die Trefferraten im Cache, wenn das Programm in einer Schleife mehrfach einen zusammenhängenden Speicherbereich von
- c1) 1024 Byte
 - c2) 2048 Byte
 - c3) 4096 Byte
- adressiert?
- d) Der oben beschriebene vollassoziative Cache werde als L1 Befehlscache benutzt. Ein Programm adressiert in einer Schleife mehrfach einen zusammenhängenden Codebereich von 2560 Bytes im Hauptspeicher. wie groß sind die Trefferraten im Cache ab dem zweiten Durchgang, wenn dieser mit ...

- d1) ... der LRU-Ersetzungsstrategie
- d2) ... der LFU-Ersetzungsstrategie
- d3) ... der FIFO-Ersetzungsstrategie

arbeitet?

- e) Ein Programm greift in einer Schleife mehrfach auf die 32-Bit-Adressen 0x00002091, 0x00005492, 0x00002093, 0x00005494, 0x0000109A, 0x0000F49B, 0x0000109C und 0x0000F49D zu. Wie sind die Trefferraten ab dem zweiten Schleifendurchlauf bei Verwendung folgender Caches (alle mit einer Cache-Zeilengänge von 16 Byte):
 - e1) vollassoziativer Cache mit 64 Zeilen?
 - e2) direkt abbildender Cache mit 64 Zeilen?
 - e3) zweifach assoziativer Cache mit 32 Sätzen zu je 2 Zeilen?
 - e4) vierfach assoziativer Cache mit 16 Sätzen zu je 4 Zeilen?

Aufgabe 2.10 (Speicherverwaltung):

In einem Speicherverwaltungssystem seien die folgenden Freispeicherbereiche verkettet in der angegebenen Reihenfolge gegeben: 4KB, 10KB, 9KB, 20KB, 19KB, 7KB. Es werden nun nacheinander Speicheranforderungen von 8KB, 14KB, 12KB, 8KB und 4KB gestellt. Beantworten Sie für die Speicherzuteilungsverfahren:

1. Rotating-First-Fit,
2. Best-Fit

die folgenden Fragen:

- a) Welche Freispeicherbereiche werden den Anforderungen zugeordnet?
- b) Wie sieht die Freispeicherliste nach Bearbeitung aller Speicheranforderungen aus?
- c) Wieviele Suchschritte in der Freispeicherliste werden durchgeführt?

Aufgabe 2.11 (Speicherverwaltung):

Ein Buddy-System zur Speicherverwaltung besitze zu Beginn einen einzigen freien Block der Länge 256KB beginnend an der Adresse 0. Die minimale Blockgröße sei 4 KB. Es nun nacheinander Speicheranforderungen von A=14KB, B=24KB, C=35KB, D=16KB und E=60KB gestellt.

- a) Welche freien Blöcke existieren nach Bearbeitung aller Speicheranforderungen?
- b) Wie hoch ist der Anteil an interner und externer Fragmentierung nach Bearbeitung aller Speicheranforderungen?
- c) Die angeforderten Blöcke mögen in der Reihenfolge A, B, C, E, D freigegeben werden. Welche Anfangsadresse und -Größe haben die freien Blöcke nach jeder Freigabe?

Aufgabe 2.12 (Dateisysteme):

Es liege ein Inode-basiertes Dateisystem mit 1 KB Blockgröße vor. Ein Inode besitze, wie im Falle von System V, 10 Einträge für direkt adressierte Blöcke. Jede Referenz auf einen Plattenblock (Blocknummer) belege ein 32 Bit-Wort.

- a) Bestimmen Sie die maximale Größe einer Datei ohne Indirektblöcke, mit 1-fach Indirektblöcken, mit 2-fach Indirektblöcken sowie mit 3-fach Indirektblöcken.
- b) Wieviele Blöcke mit Verweisinformationen benötigt das Dateisystem für eine Datei mit 10MB reinem Datenumfang zur Aufnahme der Speicherfunktion der Datei?
- c) Es wurde vorgeschlagen, dass der erste Teil jeder UNIX-Datei im selben Plattenblock gespeichert werden sollte wie sein Inode. Wie bewerten Sie diesen Vorschlag?
- d) Warum wurde der Offset-Typ von `lseek()` von `long` auf `off_t` geändert?
- e) Wie ändert sich die Lösung von (a), wenn ein Berkeley Fast File System mit 4 kB Blockgröße (und 12 Einträgen für direkt adressierte Blöcke) angenommen wird?
- f) Warum wird es im Berkeley Fast File System zugelassen, dass die Fragmente eines Blocks zu unterschiedlichen Dateien gehören können?

Aufgabe 2.13 (Dateisysteme):

Ein Plattenlaufwerk besitze 1800 Zylinder, 15 Köpfe, und 64 Sektoren je 512 Bytes je Spur. Die Umdrehungszahl sei 5400 je Minute. Die Positionierzeit sei 2.5 msec Spur-zu-Spur sowie 10 msec im Mittel. Die Datentransferrate ist mit 4 MB/sec angegeben. Das Dateisystem verwendet eine Blockgröße von 4 KB.

- a) Welche Zeit wird benötigt, um 1 MB Daten von der Platte aus aufeinanderfolgenden Blöcken zu lesen?
- b) Welche Zeit wird benötigt, wenn die Blöcke zufällig über die Platte verteilt sind?
- c) Welche Vor- und Nachteile hat das Kopieren eines Dateisystems auf physikalischer Ebene, z.B. mittels des UNIX-Dienstprogramms `dd`?