

Klausur+Musterlösung

Betriebssysteme WS 2015/16

3.02.16

Tragen Sie auf dem Deckblatt Ihre Daten in Druckbuchstaben ein.

Vorname:	
Nachname:	
Matrikelnummer:	
Studienfach:	

Ich bin im Masterstudiengang und beantrage die Beurteilung folgender Teile:

(bitte ankreuzen, was gewünscht) BS1 BS2

Bei Student*innen im **Bachelor**studiengang werden automatisch beide Teile gewertet.

Bitte tragen Sie auf jeder Seite Ihre Matrikelnummer ein und überprüfen Sie diese Klausur auf Vollständigkeit (18 Seiten!).

Verwenden Sie ausschließlich die beigelegten Blätter. Die Rückseite können Sie als Schmierpapier für Ihre Notizen verwenden. Sollten diese nicht ausreichen, so wenden Sie sich bitte an die Aufsicht. Ein Taschenrechner (kein Handy!) ist erlaubt. Andere Hilfsmittel (z. B. Handys, Bücher, eigenes Papier, etc...) sind verboten. Die Benutzung gilt als Täuschungsversuch und führt zum Ausschluss von der Klausur.

Die Klausur besteht aus zwei Teilen und enthält jeweils 8 Aufgaben mit insgesamt maximal 160 möglichen Punkten. Davon sind für eine volle Leistung (100 Prozent) in jedem Teil 60 Punkte zu erreichen. Die Anzahl der Punkte entspricht ungefähr der Bearbeitungszeit in Minuten. Für Teilnehmer im Bachelorstudiengang werden beide Teile zusammen gewertet; für Masterstudenten wird jeder Teil gesondert bewertet; ebenso die Übungspunkte.

Die maximale Bearbeitungszeit für beide Teile beträgt insgesamt 180 Minuten.

Viel Erfolg !!!

0	1	2	3	4	5	6	7	8	9	10 A	11 B	12 C	13 D	14 E	15 F
0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111

Korrekturnotizen (nicht ausfüllen!) _____

BS1								BS2								Σ	Ü	Note
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16			
10	10	8	8	10	8	12	14	12	12	14	8	10	6	12	6			

Multiple Choice-Aufgaben (Mehrfachantworten sind möglich!) 10 Punkte

1. Betriebsmittel sind ...
 - ausschließlich Hardware
 - ausschließlich Software
 - können aus Hardware oder Software bestehen
2. Was ist ein Interrupt-Vektor?
 - Array, das alle Interrupts enthält
 - Adresse der Speicherzellen für PC- und PS-Register bei einem Interrupt
 - lineare Mannigfaltigkeit im Zwischenraum
3. Eine virtuelle Maschine hat ...
 - nur eine Schnittstelle
 - zwei Schnittstellen
 - drei Schnittstellen
4. Kernel-Mode bedeutet ...
 - die Möglichkeit, Arbeitsspeicher zu segmentieren („Kerne“)
 - gibt es nur bei Unix
 - Modus in dem betriebssystem-immanente Arbeiten durchgeführt werden
 - Kernbetrieb bei Notfallsituationen (Stromausfall)
 - Betriebsmodus der vom Benutzer nicht manipuliert werden kann
5. Im Unterschied zu Threads gilt für Prozesse:
 - Sie haben einen eigenen, getrennten Speicherbereich
 - Sie benötigen keine Semaphore zur Synchronisation
 - Sie benutzen nicht dieselbe Datei
 - Sie haben unterschiedliche Prioritäten
6. Bei welcher Seitenersetzungsstrategie kann es zur Belady's Anomalie kommen?
 - LRU
 - NFU
 - FIFO
7. B-Bäume der Ordnung m ...
 - haben Blätter auf nur einer Ebene
 - haben maximal m Schlüssel pro Knoten
 - haben Knoten, die im Durchschnitt nur zur Hälfte gefüllt sind.
 - verschwenden mehr Platz in den Knoten als B^* Bäume.
8. Die mögliche Aufgaben eines Treibers sind
 - Implementierung der Applikationslogik
 - Umsetzung der virtuellen zu reellen Adressen
 - Koordination verschiedener Geräte gleichen Typs
 - Koordination der schreibenden und lesenden Prozesse
 - Pufferung der Daten

9. Eine Sitzungssemantik garantiert ...
- dass alle Änderungen an einem Dokument erhalten bleiben
 - nur die Erhaltung der letzten Änderung
 - dass alle Änderungen eines einzigen Benutzers an einem Dokument erhalten bleiben
 - nur die Erhaltung der letzten Änderung eines Benutzers
10. I/O-Mapping ...
- ist die Abbildung von I/O-Seiten in den Benutzeradressraum
 - ermöglicht schnelleren I/O
 - ist die Abbildung der Gerätereister in den Speicheradressraum
 - dient zur leichteren Programmierung

Lösungen:

1. Betriebsmittel sind...
- ausschließlich Hardware
 - ausschließlich Software
 - können aus Hardware oder Software bestehen.
2. Was ist ein Interrupt-Vektor?
- Array, das alle Interrupts enthält
 - Adresse der Speicherzellen für PC- und PS-Register bei einem Interrupt
 - lineare Mannigfaltigkeit im Zwischenraum
3. Eine virtuelle Maschine hat ...
- nur eine Schnittstelle
 - zwei Schnittstellen
 - drei Schnittstellen
4. Kernel-Mode bedeutet ...
- die Möglichkeit, Arbeitsspeicher zu segmentieren („Kerne“)
 - gibt es nur bei Unix
 - Modus in dem betriebssystem-immanente Arbeiten durchgeführt werden
 - Kernbetrieb bei Notfallsituationen (Stromausfall)
 - Betriebsmodus der vom Benutzer nicht manipuliert werden kann
5. Im Unterschied zu Threads gilt für Prozesse:
- Sie haben einen eigenen, getrennten Speicherbereich
 - Sie benötigen keine Semaphore zur Synchronisation
 - Sie benutzen nicht dieselbe Datei
 - Sie haben unterschiedliche Prioritäten
6. Bei welcher Seitenersetzungsstrategie kann es zur Belady's Anomalie kommen?
- LRU

- NFU
- FIFO

7. B-Bäume der Ordnung m ...

- haben Blätter auf nur einer Ebene
- haben maximal m Schlüssel pro Knoten
- haben Knoten, die im Durchschnitt nur zur Hälfte gefüllt sind.
- verschwenden mehr Platz in den Knoten als B* Bäume.

8. Die mögliche Aufgaben eines Treibers sind

- Implementierung der Applikationslogik
- Umsetzung der virtuellen zu reellen Adressen
- Koordination verschiedener Geräte gleichen Typs
- Koordination der schreibenden und lesenden Prozesse
- Pufferung der Daten

9. Eine Sitzungssemantik garantiert ...

- dass alle Änderungen an einem Dokument erhalten bleiben
- nur die Erhaltung der letzten Änderung
- dass alle Änderungen eines einzigen Benutzers an einem Dokument erhalten bleiben
- nur die Erhaltung der letzten Änderung eines Benutzers

10. I/O-Mapping ...

- ist die Abbildung von I/O-Seiten in den Benutzeradressraum
- ermöglicht schnelleren I/O
- ist die Abbildung der Gerätereister in den Speicheradressraum
- dient zur leichteren Programmierung

Name:

Aufgabe 2 Prozess-Scheduling (10 Punkte)

Nahezu zeitgleich treffen die Prozesse P1, P2, P3, P4, P5 in der angegebenen Reihenfolge mit den Bearbeitungsdauern $t(P1)=1$, $t(P2)=6$, $t(P3)=2$, $t(P4)=4$, $t(P5)=2$ in einem System ein, das Round-Robin für das Scheduling verwendet.

a) Zeichnen Sie Schedulingdiagramme für das Round-Robin-Schedulingverfahren und bestimmen Sie die mittleren Bearbeitungszeiten der Prozesse, falls jede Zeitscheibe

i) eine Jobeinheit lang ist, (4 Pkte)

ii) sehr kurz ist. (4 Pkte)

Nehmen Sie dabei an, dass die für einen Jobwechsel benötigte Zeit vernachlässigt werden kann.

b) Warum ergeben sich zwei unterschiedliche mittlere Bearbeitungszeiten? (2 Pkte)

Lösung:

a)

Eine Jobeinheit pro Zeitscheibe:

Alle Jobs werden hintereinander abgearbeitet mit voller Geschwindigkeit

P1 (1)															
P2 (6)															
P3 (2)															
P4 (4)															
P5 (2)															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Durchschnittliche Bearbeitungsdauer: $(1 + 15 + 7 + 13 + 9) / 5 = 45/5 = 9$ Zeiteinheiten.

Sehr kurze Zeitscheiben:

Alle Jobs werden parallel abgearbeitet, aber mit verminderter Geschwindigkeit

P1 (1)	1														
P2 (6)	1		1			1		1		1		1		1	
P3 (2)	1				1										
P4 (4)	1			1			1			1					
P5 (2)	1					1									
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Durchschnittliche Bearbeitungsdauer: $(5 + 15 + 9 + 13 + 9) / 5 = 51/5 = 10,2$ Zeiteinheiten.

b) Im ersten Fall ist die Bearbeitungszeit abhängig von der Reihenfolge, im zweiten Fall nicht. Im ersten Fall ist die Reihenfolge so, dass P1 und P3 nach Abarbeitung die anderen Jobs nicht mehr verlangsamt im Unterschied zum zweiten Fall. Wären alle Jobs vorher in absteigender Joblängen-Reihenfolge sortiert eingetroffen, so wäre dieser Effekt nicht vorhanden.

Name:

Aufgabe 3 Echtzeit-Scheduling (8 Punkte)

- a) Wie definiert sich ein Echtzeitsystem? (2 Pkte)
- b) Was ist der Unterschied des *Guaranteed Percentage-Schedulings* GPS zum dem *Rate Monotonic Scheduling* RMS? (4 Punkte)
- c) Was sind nützliche, aber nicht kritische Tasks? Geben Sie mind. zwei Beispiele. (2 Pkte)

Lösungen

- a) Ein Echtzeitsystem garantiert verbindliche Zeitschranken für jede Taskausführung. (2P)
- b) Das GPS verwendet für jeden Task eine Anzahl von Zeitscheiben, die proportional seiner CPU-Auslastung H ist. Dies garantiert eine 100%-Auslastung. (2P.) Hingegen beim RMS wird jedem Task eine Priorität gegeben, die proportional zu seiner Auftrettsfrequenz ist, was aber keine 100% Auslastung ermöglicht und bei seltenen, aber wichtigen Tasks eine Prioritätsinversion erfordert. (2P)
- c) Nützlich, aber nicht kritisch (also nicht von Zeitschranken abhängig) sind Selbsttests, Alarm-Monitore und RAM *scrubbing*. (2P)

Aufgabe 4 Prozess-Synchronisation (8 Punkte)

- a) Was versteht man unter einer „race condition“? (2 Punkte)
- b) Gegeben seien folgende Codestücke:

```
import sys
static b, e, f: Integer
var a, d, c, g: Integer

def berechne(i, j):
    b = e + j
    d = 5 + g
    g = a + i
    e = b + c + 1001 + i
    a = 1; b = 100;
    c = b + 4;
    berechne(4,5)
    print a
    print b
```

- Pseudo-Code 1 -

```
import sys
static s, d, f: Integer
var a, b, c, e, g: Integer

def berechne(j, i):
    b = e + j
    d = 5 + g
    g = a + i
    e = b + c + 102 + i
    a = 1; b = 182;
    c = g + 9;
    berechne(4,5)
    print a
    print b
```

- Pseudo-Code 2 -

Bei welchem der beiden obigen Codesegmente können *race conditions* auftreten, wenn es in mehreren Threads gleichzeitig ausgeführt wird? Begründen Sie ausführlich Ihre Antwort. (4 Punkte)

- c) Ergänzen Sie den Code geeignet, so dass *race conditions* vermieden werden. Begründen Sie Ihr Vorgehen. (2 Punkte)

Lösungen

- a) Was versteht man unter einer „race condition“?

Ein Fehler im Programmablauf tritt auf, wenn ein Prozess einen anderen überholt (*race condition*). Diese Situation kann nur entstehen, wenn beide Prozesse asynchron auf den gleichen Daten arbeiten. *Race conditions* sind deshalb besonders problematisch, da sie nur sporadisch auftreten, d. h. nicht reproduzierbar sind.

Name:

- b) Bei welchem der beiden obigen Codesegmente können *race conditions* auftreten, wenn es in mehreren Threads gleichzeitig ausgeführt wird? Begründen Sie ausführlich Ihre Antwort.

In der Aufgabe bedeutet „static“ eine Variable, die für alle Instanzen der Klasse gilt; „var“ eine Variable, die nur für die Instanz lokal gilt und auf dem Stack gehalten wird. Im Pseudo-Code 1 kann es zu *race conditions* kommen, da hier asynchron auf den globalen Variablen „b“ und „e“ gearbeitet wird. Diese Variable ist global zwischen allen Instanzen des Programms, da sie als „static“ deklariert wurde, im Unterschied zu den Variablen a, d, c, g, die „global“ nur innerhalb des Threads wirken, aber nicht von einem anderen Thread gelesen oder geschrieben werden können. Dabei ist es unerheblich, dass es durch den rekursiven Aufruf von „berechne“ ohne Abbruchbedingung irgendwann zu einem *stack overflow* kommen wird.

Zwar ist „d“ auch in Pseudo-Code 2 global, aber sie wird nur geschrieben und nie gelesen; s und f nie benutzt.

- c) Ergänzen Sie den Code geeignet, so dass *race conditions* vermieden werden. Begründen Sie Ihr Vorgehen.

Um eine *race condition* zu verhindern, setzen wir Semaphore ein. Diese sichern einen kritischen Abschnitt dahingehend ab, dass niemals zwei oder mehr Prozesse gleichzeitig diesen Code-Abschnitt durchlaufen können. Dazu ergänzen wir den Pseudo-Code 1 folgendermaßen:

```
import sys
static b, e, f, s: Integer
var a, d, c, g, p: Integer

def berechne(i, j):
    P(s)
    d = e + j
    g = 5 + g
    g = a + i
    e = b + c + 1001 + i
    a = 1; b = 100;
    c = b + 4;
    p = b; // ergänzt
    V(s)
    berechne(4, 5)
    print a
    print p // geändert
    .....
```

- Pseudo-Code 1 -

Die Operation P(s) muss vor Verwendung von b und e kommen, da hier lesend und schreibend zugegriffen wird, und V(s) muss noch vor dem rekursiven Aufruf kommen, um zu verhindern, dass der selbe Thread sich nach der Rekursion blockiert.

Die Aufgabe ist allerdings so nicht lösbar, da V(s) eigentlich nach „print b“ kommen müsste, da hier das eigene Ergebnis ausgegeben wird, was vorher nicht verändert werden sollte. So aber ist nur der Schutz der Operationen innerhalb des kritischen Abschnitts möglich, und print b ist nicht geschützt. Um auch das richtige Ergebnis zu Drucken zu erreichen, müsste im kritischen Abschnitt vorher mit p = b das Ergebnis etwa einer lokalen Variablen g zugewiesen werden, die dann anstelle von b ausgedruckt wird. Dier Aufgabe ist also nur mit einer Codemodifikation lösbar.

Name:

Aufgabe 5 Verklemmungen: Der Banker-Algorithmus (10 Punkte)

1. Nennen Sie die vier Bedingungen, bei denen eine bei Nichterfüllung ausreicht, um Verklemmungen unmöglich werden zu lassen. **(4 Punkte)**

2. Testen Sie mit Hilfe des Banker-Algorithmus folgenden Fall: **(6 Punkte)**
Noch benötigte Betriebsmittel

Prozesse					
P1	2	0	5	1	0
P2	0	1	1	0	0
P3	1	0	4	0	0
P4	3	2	6	1	2
P5	1	0	1	1	0

Schon belegte Betriebsmittel

P1	0	1	1	2	2
P2	2	2	1	1	1
P3	2	1	0	1	2
P4	4	1	3	2	1
P5	0	0	3	0	0

Frei	2	0	2	1	0
-------------	---	---	---	---	---

Sollte eine Verklemmung entstehen, lösen Sie diese auf. Schreiben Sie ihren Rechenweg nachvollziehbar auf.

Lösungen

a) Nennen Sie die vier Bedingungen, bei denen eine bei Nichterfüllung ausreicht, um Verklemmungen unmöglich werden zu lassen. **(4 Punkte)**

- mutual exclusion (Beschränkte Belegung)
- hold and wait (Zusätzliche Belegung)
- no preemption (Keine vorzeitige Rückgabe)
- circular wait (Gegenseitiges Warten)

b) Wir starten mit dem Test, ob P1 abschließen kann. Dies ist nicht der Fall, bis die Reihe an P5 kommt.

Frei = 2 0 2 1 0

Darauf kann Prozess **P5** abschließen und gibt zurück 0 0 3 0 0, →Frei = 2 0 5 1 0

Wir starten erneut mit P1.

Diesmal kann **P1** abschließen und gibt zurück 0 1 1 2 2, →Frei = 2 1 6 3 2

Wir starten die Liste erneut und testen mit P2.

Diesmal ist **P2** erfolgreich, kann abschließen und gibt 2 2 1 1 1, →Frei = 4 3 7 4 3

Erneutes Starten der Liste testet **P3**, schließt ab und gibt 2 1 0 1 2 →Frei = 6 4 7 5 5

So muss nur noch **P4** getestet werden, gibt 4 1 3 2 1 →Frei = **10 5 10 7 6**

Damit geht der Banker-Algorithmus zu Ende und alle Prozesse sind markiert.

Es liegt also **keine Verklemmung** vor.

Es gibt auch diverse andere mögliche Lösungsabläufe, falls die Liste nicht immer nach dem erfolgreichen Markieren eines Prozesses von Anfang an durchsucht wird.

Name:

Aufgabe 6 Speicherverwaltung: Buddy-Systeme (8 Punkte)

Ein Computer benutzt das Buddy-System zur Speicherverwaltung. Zu Beginn verfügt er über einen freien Speicherblock von 1 GiB, beginnend bei der Adresse 0.

Geben Sie jeweils ein Abbild der Speicherbelegung nach jedem Speicherzugriff, wenn die folgenden Speicheranforderungen (*request*) und Freigaben (*release*) in der angeführten Reihenfolge bedient werden:

A: request 64MiB, B: request 230MiB, C: request 100MiB, D: request 129MiB, release A, release B, E: request 75MiB, release C, release E, release D

Beschriften Sie dabei die zusammenhängenden Speicherblöcke jeweils mit ihrer Größe.

Lösung

1024					
A	64	128	256	512	
A	64	128	B 230	512	
A	64	C 100	B 230	512	
A	64	C 100	B 230	D 129	256
128	C 100	B 230	D 129	256	
128	C 100	256	D 129	256	
E75	C 100	256	D 129	256	
E75	128	256	D 129	256	
512			D 129	256	
1024					

Aufgabe 7 Virtueller und reeller Speicher (12 Punkte)

Gegeben seien die folgenden 4 virtuellen 16-Bit Speicheradressen in Hexadezimaldarstellung:

- 75B4
- 8AC6
- 1E9C
- 5B3E

Die Adresse ist von links nach rechts kodiert. Das erste Bit wird nicht benutzt. Die nächsten 3 Bits kodieren den jeweiligen Index in der Basis-Seitentabelle, die weiteren 3 Bits den Index in der entsprechenden Tafel. Die übrigen 9 Bits bilden den *Offset*.

			Index 0				Index 1			Offset					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Bestimmen Sie anhand der folgenden Abbildung die zu a), b), c) und d) gehörenden physikalischen 32 Bit-Adressen. Geben Sie diese in Hexadezimaldarstellung an.

Die physikalische Speicheradresse der Basis-Seitentabelle ist 9A38B75A.

Name:

7	NIL
6	39BE
5	B3C7
4	5B4C
3	NIL
2	1A75
1	87D3
0	A380

B47AB75A

7	B47AB75A
6	NIL
5	5A37B75A
4	A3EBB75A
3	B47AB75A
2	A3EBB75A
1	A3EBB75A
0	5A37B75A

9A38B75A

7	NIL
6	B125
5	4CC1
4	3333
3	B97C
2	NIL
1	5E4A
0	NIL

5A37B75A

7	NIL
6	D2A6
5	8AC6
4	2BE9
3	NIL
2	D17F
1	NIL
0	7A34

A3EBB75A

Lösung

Je Aufgabe sind 3 Operationen (je 1 Pkt) durchzuführen

a) 75B4 → 0111 0101 1011 0100 (1Pkt)

→ 7 2 → 1A75 → 0 001.1 010.0 111.0 1011 1011 0100 (1Pkt)

Ergänzung für 32 Bit mit führenden Nullen → 0 0 3 4 E B B 4 (1Pkt)

b) 8AC6 → 1000 1010 1100 0110 (1Pkt)

→ 0 5 → 4CC1 → 0 100.1 100.1 100.0 0010 1100 0110 (1Pkt)

Ergänzung für 32 Bit mit führenden Nullen → 0 0 9 9 8 2 C 6 (3 Punkte) (1Pkt)

c) 1E9C → 0001 1110 1001 1100 (1Pkt)

→ 1 7 → **Fehler!** Die Adresse existiert nicht (3 Punkte) (2Pkt)

d) 5B3E → 0101 1011 0011 1110 (1Pkt)

→ 5 5 → 4CC1 → 0 100.1 100.1 100.0 0011 0011 1110 (1Pkt)

Ergänzung für 32 Bit mit führenden Nullen → 0 0 9 9 8 3 3 E (3 Punkte) (1Pkt)

Aufgabe 8 Thrashing (14 Punkte)

- Wodurch kommt Thrashing zustande? Was ist die entscheidende Bedingung dafür? (4 Pkte)
- Welche Maßnahmen kennen Sie, um Thrashing zu verhindern oder zu vermindern? (4 Pkte)
- Was sind die Unterschiede zwischen dem Workings set-Modell, dem *page fault frequency*-Modell und dem Nutzungsgradmodell? (6 Punkte)

Lösung

- Wodurch kommt Thrashing zustande? Was ist die entscheidende Bedingung dafür? (4 Pkte)

Wenn die Zeit zum Seitenwechseln (paging) länger wird als die Zeit zur Bearbeitung des Jobs, so wird die Gesamtbearbeitungszeit der Jobs nur von den Seitenwechselzeiten bestimmt. Dies führt zu deutlich längerer Bearbeitungszeit der Jobs, das System „quält“ sich mit hoher Plattenaktivität.

Die Bedingung dafür ist also „Seitenwechselzeit \geq Jobbearbeitungszeit“. (4 Pkte)

- Welche Maßnahmen und Modelle kennen Sie, um Thrashing zu verhindern oder zu

Name:

vermindern? (4 Pkte)

Hardwaremaßnahmen: Weitere Festplatten fürs Swappen, SSD-verwenden, große Seitenrahmen festlegen. (2 Pkte)

Programmiermaßnahmen: Inline-Code, lokale Variablen verwenden. (2 Pkte)

BS-Maßnahmen: (1Pkt)

- c) Was sind die Unterschiede zwischen dem Working set-Modell, dem *page fault frequency*-Modell und dem Nutzungsgradmodell? (6 Punkte)
- Prozessmanagement nach dem **Working set-Modell** füllt den Speicher nur mit so vielen Prozessen, dass deren vorher festgelegten *working sets* hineinpassen. (2Pkt)
 - Beim **Page Fault Frequency-Modell** wird ein Prozess stillgelegt, wenn die Seitenfehlerrate einen vorher festgelegten Schwellwert überschreitet oder wieder in die *bereit*-Schlange einsortiert, wenn die Rate unterhalb des Schwellwerts ist. (2 Pkte)
 - Beim **Nutzungsgradmodell** wird zusätzlich zur der Seitenfehlerrate auch die CPU-Arbeitszeit (Idle-Zeit) betrachtet und die gewichtete Gesamtsumme als Indikator gewertet, der unterhalb eines Schwellwerts bleiben soll (Arbeitspunkt auf der ansteigenden Kennlinie). (2 Pkte)

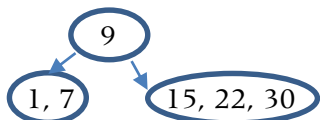
Ende BS1 - Klausur Teil 1

Name:

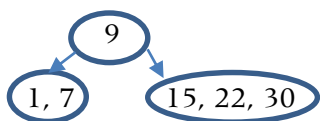
Beginn BS2 - Klausur Teil 2

Aufgabe 9 Dateiverwaltung: B- und B*-Bäume (12)

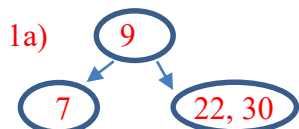
- Zeichnen Sie den Baum, der sich für $m=4$ (4 Verzweigungen pro Container) nach Einfügen der Schlüssel 7, 22, 9, 30 ergibt
 - in die leere Wurzel eines B-Baumes
 - in die leere Wurzel eines B*-Baumes
- Zeichnen Sie den Baum, der sich für $m = 4$ nach Einfügen der Schlüssel 16 und 3 ergibt
 - in folgenden B-Baum



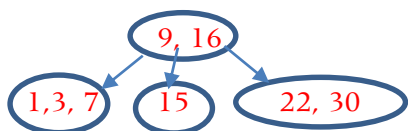
- in folgenden B*-Baum



Lösung

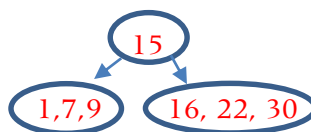


- 2a) B-Baum: Container zerteilen

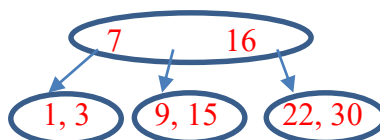


und mit 3 auffüllen.

- b) B*-Baum: mit Schlüssel 16 überfließen



und beim Einfügen von Schlüssel 3 in B*-Baum: Container zusammenfassen und in drei zerteilen



Name:

Aufgabe 10 Implementierung Dateisysteme (12 Punkte)

- a) Erklären Sie, wie mittels i-nodes sowohl sehr kleine als auch riesig große Dateien effizient verwaltet werden können. (8 Punkte)
- b) Angenommen, die Tabellen enthalten je 20 Verweise, wie viele Seiten können durch eine i-node-Datenstruktur 3. Ordnung maximal verwaltet werden? (4 Punkte)

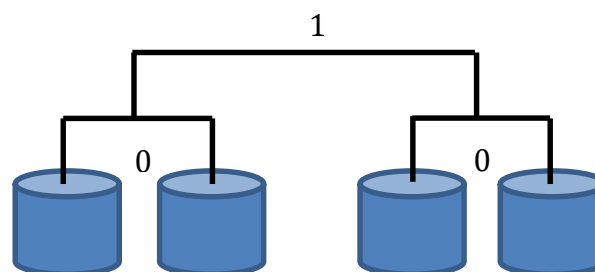
Lösung

- a) Eine i-node enthält die ersten 10 Adressen (1Pkt) einer Datei. Dadurch können Dateien, die höchstens 10 Blöcke belegen (davon gibt es i.A. sehr viele, z.B. Verzeichnisse) sehr schnell bearbeitet werden. Des Weiteren gibt es drei weitere Adressen von Blöcken (1Pkt) für die einfach-, doppelt- und dreifach-indirekte Adressierung(1Pkt). Dabei werden Tabellen mit jeweils 256 Verweisen adressiert. Je nach Grad der Indirektion verweisen die Tabelleneinträge auf Datenblöcke oder weitere Indextabellen (1Pkt). Bei einer (sehr großen) Datei stehen die Adressen der ersten 10 Blöcken in der i-node, (1Pkt) die nächsten 256 Blöcke in der Tabelle, (1Pkt) deren Adresse unter einfach indirekt vermerkt ist, dann kommen die zweifach- (1Pkt) und schließlich die dreifach-indirekt adressierten Blöcke. (1Pkt)
- b) $10+20+20^2+20^3$ (je 1 Pkt pro richtiger Zahl) = $30+400+8000 = 8430$ Blöcke

Aufgabe 11 RAID (14 Punkte)

Die Wahrscheinlichkeit, dass eine Festplatte in einem gegebenen Zeitraum einen Fehler (Head-Crash) aufweist betrage p .

- a) Bestimmen Sie die Wahrscheinlichkeit, dass im Beobachtungszeitraum Daten verloren gehen, für eine Konfiguration mit zwei gleichen Platten für ein RAID-0-System. (4 Punkte)
- b) Bestimmen Sie die Wahrscheinlichkeit, dass im Beobachtungszeitraum Daten verloren gehen, für eine Konfiguration mit vier gleichen Platten für ein RAID-01-System, wobei jeweils zwei RAID-0 Platten mittels *striping* zu einem Subsystem verbunden sind und beide Subsysteme mittels RAID 1 gespiegelt werden. (10 Punkte)



Lösung

- a) Bei 2 Platten D1 und D2 gibt es im RAID0-Modus keine Fehlertoleranz. Das Gesamtsystem fällt also aus, wenn eine der Platten oder beide defekt sind. Die Wahrscheinlichkeit dafür kann auf zwei Arten berechnet werden:
- Zum einen direkt über die Summe aller Ausfallmöglichkeiten, also **P(Ausfall) = P(Ausfall D1) + P(Ausfall D2) + P(Ausfall beider) = $2(1-p)p + p^2 = 2p - 2p^2 + p^2 = 2p - p^2$.**

Name:

- Zum anderen über die Gegenwahrscheinlichkeit $P(\text{ok})$ mit $P(\text{Ausfall}) = 1 - P(\text{ok})$. Da bei RAID0 das System nur dann ok ist, wenn beide Platten ok sind, ist bei unabhängigen Platten die Auftretenswahrscheinlichkeit die Produktwahrscheinlichkeit $P(\text{ok}) = (1-p)(1-p)$ und damit

$$P(\text{Ausfall}) = 1 - (1-p)(1-p) = 1 - [1-2p+p^2] = 2p - p^2.$$

Begründung richtig: 1Pkt, Formel ok: 2 Pkte, Endformel ausgerechnet: 1 Pkt.

- b) Bei vier Platten ist das schwieriger direkt hinzuschreiben. Das Problem lässt sich verschieden angehen.

- Die eine Idee besteht darin, beide RAID0-Kombinationen als Subsysteme aus je einem System wie in Aufgabe a) aufzufassen und weiterhin die Information zu verwenden, dass RAID1-Systeme genau dann ausfallen, wenn beide Subsysteme ausgefallen sind. Sind beide Subsysteme unabhängig voneinander, so ist die Auftretenswahrscheinlichkeit die Produktwahrscheinlichkeit, also $P(\text{Ausfall}) = P(\text{Ausfall Subsystem A}) P(\text{Ausfall Subsystem B})$. Einsetzen der Ergebnisse von a) ergibt

$$P(\text{Ausfall}) = (2p - p^2)(2p - p^2) = p^4 - 4p^3 + 4p^2$$

- Eine andere Idee ist, wieder die Gegenwahrscheinlichkeit $P(\text{ok})$ einzusetzen. In diesem Fall ist $P(\text{Ausfall}) = 1 - P(\text{ok})$ und $P(\text{ok})$ ist die Summe aller Fälle, in denen das System funktioniert. Dies ist der Fall, wenn kein oder nur ein Subsystem ausgefallen ist:

$$\begin{aligned} P(\text{ok}) &= P(\text{ beide Subsysteme ok}) + P(\text{Subsystem 1 ok}) + P(\text{Subsystem2 ok}) \\ &= (1-p)^2(1-p)^2 + (1-p)^2 [1-(1-p)^2] + [1-(1-p)^2](1-p)^2 \\ &= (1-2p+p^2)(1-2p+p^2) + (1-2p+p^2)(2p-p^2) + (2p-p^2)(1-2p+p^2) \\ &= 1 - 4p^2 + 4p^3 - p^4 \end{aligned}$$

Damit ist der Ausfall als Gegenwahrscheinlichkeit

$$\begin{aligned} P(\text{Ausfall}) &= 1 - P(\text{ok}) \\ &= 1 - [(1-p)^2(1-p)^2 + (1-p)^2 [1-(1-p)^2] + [1-(1-p)^2](1-p)^2] \\ &= 1 - (1 - 4p^2 + 4p^3 - p^4) = 4p^2 - 4p^3 + p^4 \end{aligned}$$

- Eine sichere, aber umfangreichere Lösung besteht darin, indem man eine Tabelle aller möglichen Zustände der vier Festplatten D1, D2, D3 und D4 aufstellt und die Wahrscheinlichkeiten aller Zustände aufsummiert, die zu Datenverlusten führen.

Dabei berücksichtigen wir die drei Fälle, die zum Ausfall führen: den Ausfall von 4 Einheiten, den Ausfall von 3 Einheiten, und den Ausfall von 2 Einheiten, wenn sie in unterschiedlichen Subsystemen sind.

D1	D2	D3	D4	Ausfall prob.	Gesamt ausfall
0	0	0	0	p^4	true
0	0	0	1	$p^3(1-p)$	true
0	0	1	0	$p^3(1-p)$	true
0	0	1	1	$p^2(1-p)^2$	false
0	1	0	0	$p^3(1-p)$	true
0	1	0	1	$p^2(1-p)^2$	true
0	1	1	0	$p^2(1-p)^2$	true
0	1	1	1	$p(1-p)^3$	false

Name:

1	0	0	0	$p^3(1-p)$	true
1	0	0	1	$p^2(1-p)^2$	true
1	0	1	0	$p^2(1-p)^2$	true
1	0	1	1	$p(1-p)^3$	false
1	1	0	0	$p^2(1-p)^2$	false
1	1	0	1	$p(1-p)^3$	false
1	1	1	0	$p(1-p)^3$	false
1	1	1	1	$(1-p)^4$	false

Damit ergibt sich die Ausfallwahrscheinlichkeit als Summe von 9 Fällen aus der Tabelle mit „true“ zu

$$\begin{aligned} P(\text{Ausfall}) &= p^4 + 4p^3(1-p) + 4p^2(1-p)^2 = p^4 + 4p^3 - 4p^4 + 4p^2(1-2p+p^2) \\ &= p^4 + 4p^3 - 4p^4 + 4p^2 - 8p^3 + 4p^4 = \mathbf{p^4 - 4p^3 + 4p^2} \end{aligned}$$

- Wir können natürlich auch in diesem Fall wieder die Gegenwahrscheinlichkeit einsetzen. Es ist $P(\text{ok})$ die Summe aller Fälle, bei denen das Gesamtsystem ok ist. Dies sind in der Tabelle alle 7 Fälle mit „false“, so dass sich ergibt

$$\begin{aligned} P(\text{ok}) &= p^2(1-p)^2 + p(1-p)^3 + p(1-p)^3 + p^2(1-p)^2 + p(1-p)^3 + p(1-p)^3 + (1-p)^4 \\ &= 2p^2(1-p)^2 + 4p(1-p)^3 + (1-p)^4 \\ &= 2p^2(1-2p+p^2) + 4p(1-p)(1-2p+p^2) + (1-2p+p^2)(1-2p+p^2) \\ &= (2p^2 - 4p^3 + 2p^4) + (-12p^2 + 12p^3 - 4p^4) + (1 + 6p^2 - 4p^3 + p^4) \\ &= (1 - 4p^2 + 4p^3 - p^4) \end{aligned}$$

Damit ist der Ausfall als Gegenwahrscheinlichkeit

$$\begin{aligned} P(\text{Ausfall}) &= 1 - P(\text{ok}) \\ &= 1 - (1 - 4p^2 + 4p^3 - p^4) = \mathbf{4p^2 - 4p^3 + p^4} \end{aligned}$$

Begründung richtig: 1Pkt, Formel ok: 8 Pkte, Endformel ausgerechnet: 1 Pkt.

Aufgabe 12 Schattenserver und NC (8 Punkte)

- Wie funktioniert ein Netzwerk mit Schattenserver (SS), und wie ein Netzwerk mit Netzwerk-Computern NC? (3 Pkte)
- Was unterscheidet einen Client im SS-Netz von einem Client im NC-Netz funktionell, hardware- und softwaremäßig? (3 Pkte)
- Welche Vorteile bietet die jeweilige Konfiguration? (2 Punkte)

Lösungen

- In einem Netzwerk mit Schattenserver enthält der Server alle Dateien, die auch auf den Client-Rechnern sind (1Pkt) und bildet damit einen back-up aller im Netz befindlichen Systeme. Alle Client-Rechner sind aber autonom und können auch ohne Netzwerk arbeiten. (1Pkt)
Im NC-Netzwerk laden alle NC-Computer Programme und Daten direkt vom Server; ohne Netzwerk können sie nicht funktionieren. (1Pkt)
- Ein Client-Rechner im SS-Netz ist autonom und ist sowohl hardwaremäßig, programmäßig als auch datenmäßig vollständig ausgestattet. (1Pkt) Das Netzwerk dient allein der Aktualisierung der Software und der Daten. (1Pkt) Im Unterschied dazu haben die NC-Rechner keine Festplatten etc., da alle Daten und Programme nur auf dem Server gespeichert sind. (1Pkt)

Name:

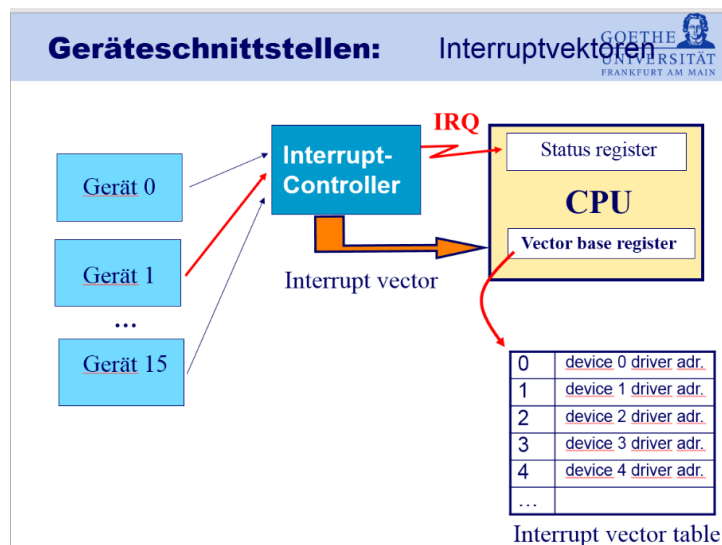
- c) Die Vorteile des NC sind leichte Skalierbarkeit, Datenkonsistenz, leichtere Wartbarkeit, Energieeinsparung und preiswerte Clients. (1Pkt) Die Vorteile des SS sind auch leichte Skalierbarkeit, Datenkonsistenz, leichtere Wartbarkeit, aber auch Fehlertoleranz und Autonomie bei Netzausfall. Dafür sind sie teurer. (1Pkt)

Aufgabe 13 Treiber und I/O-Mapping (10 Punkte)

- a) Was ist ein Treiber und was der Unterschied zwischen einer abstrakten Maschine, einem Treiber und einem Gerätetreiber ? (4 Punkte)
- b) In welchem Systemmodus laufen Treiber? Warum? (1 Punkt)
- c) Was ist *I/O-mapping* und wozu dient es? (2 Punkte)
- d) Wie wird der dazu gehörende Treiber bei dem Interrupt eines Geräts ausgewählt? (3 Punkte)

Lösungen

- a) Ein Treiber ist eine Zwischenschicht im Betriebssystem zwischen den Anforderungen und der Ein- und Ausgabe. (2 Pkt) Dies entspricht einer virtuellen Maschine und hat 2 Schnittstellen. Im Gegensatz dazu betrachtet eine abstrakte Maschine ein Gerät von außen und hat deshalb nur eine Schnittstelle. Ein Beispiel dafür ist eine logische Einheit. (1 Pkt) Ein Gerätetreiber enthält gerätetypische Merkmale und Dienste, im Unterschied etwa zu einem Dateitreiber, der nicht an ein bestimmtes Gerät gebunden ist. (1 Pkt)
- b) Kernel-Mode, um möglichst rasch und privilegiert auf alle Hardwareadressen und Daten zugreifen zu können. (1Pkt)
- c) Beim *I/O-mapping* werden die Gerätereister wie Speicher an bestimmten Adressen im Adressraum angesprochen. Damit wird die Benutzung von speziellen, hardwareabhängigen I/O-Maschinenbefehlen vermieden. (2 Punkte)
- d) Der Interrupt liefert durch Hardware die Interruptnummer (1Pkt) (*Interruptvektor*(1Pkt)), der einem Index in einer Interrupt-Tabelle entspricht. In diesem Tabelleneintrag ist die Treiberadresse gespeichert, (1Pkt) die dann benutzt wird, um den Treiber anzuspringen.



Name:

Aufgabe 14 Remote Procedure calls (6 Punkte)

- a) Welche Vorteile haben RPC? Wozu kann man sie in verteilten Systemen nutzen? (2 Punkte)
b) Gegeben sei eine 16-Bit-Ganzzahl, die wie folgt in einem Speicher abgelegt sei:

Speicheradresse	0x0	0x1
Inhalt(hexadezimal)	00	2A

Als welche Zahl wird der Speicherinhalt (also die an Adresse 0x0 startende 16-Bit-Zahl) von einem Prozessor interpretiert, dessen Bytereihenfolge

- (1) Little Endian
(2) Big Endian

ist?

Geben Sie die Zahl jeweils in Hexadezimal- sowie Dezimaldarstellung an. (4 Punkte)

Lösungen

- a) RPC können Dienstleistungen anderer Rechner (Server) erbringen. (1 Pkt) Dies kann zu mehr Fehlertoleranz (Auswahl verfügbarer Dienste), zu mehr Leistungsverteilung (leistungsstarke Server), zu mehr Funktionalität (Dienste von Spezialrechnern, etwa Farblaser-ausdruck) und zu mehr Datenkonsistenz (zentraler file server) führen. (1 Pkt)
b) Die Umrechnungen ins Dezimalsystem gehen wohl am schnellsten über das Binärsystem:
(1) Little Endian: Byte 0 steht rechts (niederw.) $2A\ 00_{16} = 10\ 1010\ 0000\ 0000_2 = 10.752_{10}$
(2) Big Endian: Byte 0 steht links (hochwertig) $00\ 2A_{16} = 10\ 1010_2 = 42_{10}$

Die Umrechnung in b) ins Dezimalsystem ist zwar etwas aufwendiger (aber viel kopfreundenfreundlicher geht es leider nicht, haben extra schon 16 Bit statt 32 Bit genommen), aber durch sukzessives Verdoppeln sollte man relativ schnell auf $512 + 2.048 + 8.192 = 10.752$ kommen. Oder man nutzt seinen mitgebrachten Taschenrechner. (4 Punkte)

Hexzahl richtig:(1 Pkt), Dezimalzahl richtig: (1 Pkt)

Aufgabe 15 Malware – Angriffe (12 Punkte)

- a) Wie lauten die drei wichtigen Authentifizierungsarten? Nennen Sie jeweils mindestens zwei Beispiele (6 Punkte)
b) Welche Angriffsarten von *malware* kennen Sie für Ring 3, 2, 1, 0, -1, -2, -3 - Angriffe? (6 Pkte)

Lösungen

- a) „Etwas, was Sie wissen“ : Passwort, PIN, Testfragen (2 Pkt)
„Etwas, was Sie haben“ : Kreditkarte, Handy, Token-Generator (2 Pkt)
„Etwas, was Sie sind“ : Fingerabdruck, Iris, Bio-Daten (2 Pkt)

- b) Ring 3: **user mode rootkits**: Fälschung von Nutzerbibliotheken
Ring 2: shared libs: Fälschung von Systembibliotheken
Ring 1: system calls
Ring 0: **kernel mode rootkits**: falsche Treiber, Fälschung von system calls
Ring -1: hypervisor rootkits (z.B. blue pill): Einschleusen einer Hypervisor VM zwischen HW und BS
Ring -2: **SMM rootkits**: system management mode (Intel x86): Eigenes RAM etc.
Ring -3: Mainboard vPro MCH chipset (Intel Active Management Technology AMT): eigener RISC Prozessor mit Flash-RAM/ROM) für *remote administration*: Fälschung von Administrator-Software
(je 1 Punkt)

Name:

Aufgabe 16 ACL und Rollen (6 Punkte)

- a) Wie funktionieren *access control lists* ACL und wie das *capability-oriented model* COM? (2 Punkte)
- b) Was sind die Vorteile von ACL und von COM? (2 Punkte)
- c) Welche Vorteile bietet darüber hinaus das Rollen-Modell? (2 Punkte)

Lösungen

- a) Für jedes Objekt gibt es eine ACL, in der die Zugriffsrechte einzelner Benutzer und Benutzergruppen stehen. Jede(r) Nutzer/in hat auch eine COM-Liste, in der für ihn/sie alle Berechtigungen aller Objekte aufgeführt sind.
- b) Damit ist es möglich, bei jeder ACL sehr feine und genaue Zugriffsrechte aufzuführen. Dies gilt auch für COM. Der Unterschied liegt im Speicherort und in der Zugriffsgeschwindigkeit: Gibt es sehr viele Objekte, so ist es einfacher, sie in einer COM des Nutzers aufzuführen, als bei jedem Objekt eine ACL zu speichern.
- c) Das Rollenmodell bietet darüber hinaus die Möglichkeit, die Zugriffsbeschränkungen von einem konkreten Nutzer zu abstrahieren und sie nur einmal für einen generischen Benutzer und seine möglichen Arbeitsaktivitäten festzulegen. Zum einen spart man sich so, gleiche Rechte auf die COM von Nutzern mit gleichem Arbeitsauftrag zu kopieren, zum anderen ist es auch leichter, die Rechte zu ändern, wenn der Arbeitsauftrag sich ändert.