

Vorlesung Betriebssysteme WS 2013/2014

Aufgabenblatt 4 vom 9. Januar 2014

(Vorstellung der Lösungen bei den Tutoren bis zum 23.01.14)

Aufgabe 4.1: (15 Punkte)

Erläutern Sie Ihrem Tutor folgende Begriffe und deren Verwendung im Kontext eines Betriebssystems!

- Virtueller Speicher
- Paging, Page-Fault, Page-Replacement Algorithmus
- Swapping, Interne/Externe Fragmentierung

Aufgabe 4.2: (30 Punkte)

Schreiben Sie Programmteile in der Programmiersprache C, welche Speicherverwaltung mit verschiedenen Algorithmen zur Ausführung des bereitgestellten Programmrahmens unter Linux. Die Auswahl des Algorithmus zur Speicherverwaltung erfolgt mit Hilfe einer Präprozessorvariable (CFLAGS):

- FIRST First Fit
- BUDDY Buddy Algorithmus

Folgende Funktionen werden durch den Programmrahmen gerufen und müssen für jeden Algorithmus implementiert werden. Wägen Sie eine minimale Blockgröße ausgehend von der verwalteten Speichergröße für den Buddy Algorithmus ab. Die Verwaltungsstrukturen für den Speicherbereich sind freigestellt und aber nicht in diesem untergebracht werden:

```
void* bs_malloc(size_t size);  
void bs_free(void* ptr);
```

Der verfügbare Speicherbereich ist fest im Programmrahmen vorgegeben und darf nicht verändert werden. Die Fehlerbehandlung soll wie bei `malloc(3)` bzw. `free(3)` erfolgen. Stellen Sie die Mehreintrittsfähigkeit der implementierten Funktionen sicher. Erläutern Sie Ihrem Tutor Ihre Implementierung und führen Sie Ihr Programm vor! Erstellen Sie ein Makefile für Linux mit den Zielen `aufg42_first` und `aufg42_buddy`, die Programme mit dem Zielnamen erzeugen. Verpacken Sie alle Source-Code Dateien (inklusive Makefile) in eine ZIP-Datei (`blatt42.zip`). Diese ZIP-Datei muss mindestens 24 Stunden vor dem Tutoriumstermin in das Abgabesystem eingestellt werden.

Aufgabe 4.3: (25 Punkte)

Ein Prozess benutzt Speicherseiten, die durch die folgende Zugriffsreihenfolge in verfügbare Seitenrahmen im Hauptspeicher geladen werden. Alle Seitenrahmen sind zu Beginn leer.

1 0 7 4 1 0 5 1 0 7 4 5

- a) Es stehen 3 Seitenrahmen zur Verfügung
- b) Es stehen 4 Seitenrahmen zur Verfügung

Stellen Sie ihrem Tutor Diagramme für die Seitenersetzungsstrategien FIFO und LRU vor, welches die Belegung der Seitenrahmen zu jedem Zeitpunkt zeigt. Markieren Sie Seitenzugriffsfehler und Berechnen das Trefferverhältnis. Welches Verhalten lässt sich beim Vergleich der Seitenersetzungsstrategien beobachten?

Aufgabe 4.4: (15 Punkte)

Gegeben sei ein 36-Bit-Rechnersystem.

- Die Seitengröße (*Page-Frame Size*) beträgt 16 KiB.
- Der Prozessor unterstützt maximal 64 GiB physikalischen Speicher.
- Die virtuelle Speicherverwaltung unterstützt lediglich eine Ebene von Seitentabellen, d.h. zu jedem Prozess gehört genau eine Tabelle, die den gesamten virtuellen Speicher auf den physikalischen abbildet. Diese Tabellen können nicht ausgelagert werden.
- Der Rechner hat insgesamt 1 GiB physikalischen Speicher. Das Betriebssystem benötigt mindestens 128 MiB Speicher, der nicht ausgelagert werden kann. In diesem Speicherbereich sind evtl. Seitentabellen noch nicht eingerechnet.

Beantworten Sie Ihrem Tutor folgende Fragen:

- Wie viele Einträge enthält eine Seitentabelle?

- Wie viel physikalischer Speicher wird von einer Seitentabelle belegt? Jeder Seitentabelleneintrag soll die Seitennummer und 10 Steuerungs-Bits enthalten.
- Wenn drei Prozesse in diesem System laufen, wie groß kann das *Working-Set* eines Prozesses maximal sein?

Aufgabe 4.5: (15 Punkte)

Führen Sie das InstantLab-Experiment **LeakyApp** aus und beantworten Sie die Fragen in der Experimentbeschreibung auf der Vorlesungsseite.

Erläutern Sie Ihrem Tutor die Rollen der *Standby*-, *Modified*-, *Free*- und *Zero page*- Listen in der Windows Speicherverwaltung!

Welche Eigenschaften haben Seiten, die in den einzelnen Listen enthalten sind?

Musterlösung Aufgabenblatt 3

Aufgabe 4.1 (15):

Alle Gruppenmitglieder wissen gleich gut Bescheid:

3

Aufgabe 4.2 (30):

Vorführung/Programm erklären/Kommentare usw.

5

Funktionalität

- Algorithmen korrekt implementiert
- Programme funktionieren
- Fehlerbehandlung

10

5

3

Formalitäten

- Korrektes Makefile
- Pünktliche Abgabe

2

2

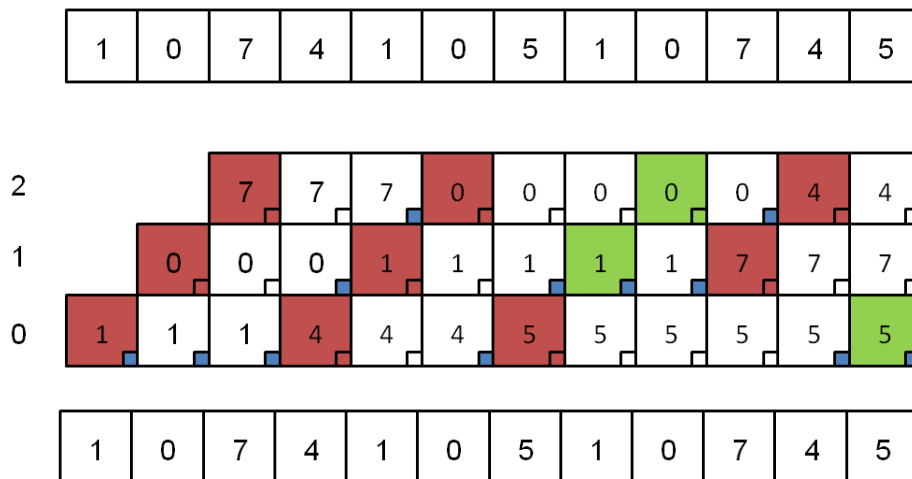
Alle Gruppenmitglieder wissen gleich gut Bescheid:

3

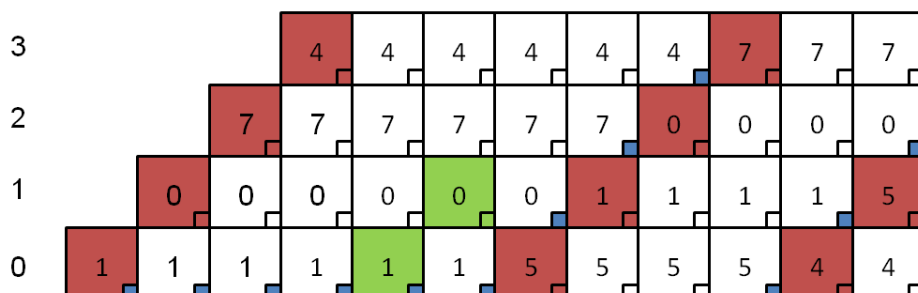
Aufgabe 4.3 (25)

- korrekte Darstellung der Speicherbelegung zu jedem Zeitpunkt 5
- FIFO 5
- LRU 5
- Pagefaults markiert 5
- FIFO Anomalie (Belady) *An Anomaly in Space-Time Characteristics of Certain Programs Running in a Paging Machine* erkannt 2
- Alle Gruppenmitglieder gleich gut informiert 3

FIFO



LRU



1	0	7	4	1	0	5	1	0	7	4	5
---	---	---	---	---	---	---	---	---	---	---	---

2			7	7	7	0	0	0	0	0	5
1		0	0	0	1	1	1	1	1	4	4
0	1	1	1	4	4	4	5	5	5	7	7

1	0	7	4	1	0	5	1	0	7	4	5
---	---	---	---	---	---	---	---	---	---	---	---

3				4	4	4	4	4	4	7	7	7
2			7	7	7	7	5	5	5	5	4	4
1		0	0	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	1	1	1	1	5

Aufgabe 4.4 (15):

Wie viele Einträge enthält eine Page-Table?

64 GByte Gesamtspeicher / 16KByte Seitengröße = $2^{22} = 4.194.304$ Einträge

4

Wie viel physikalischer Speicher wird von einer Seitentabelle belegt? Jeder Seitentableneintrag soll die Page-Frame Nummer und 10 Kontroll-Bits enthalten.

(22 Bit für Page-Frame + 10 Steuerungs-Bits) * 2^{22} Einträge =

32 Bit * 4.194.304 Einträge = 134.217.728 Bits = 16 MiB

4

- Wenn drei Prozesse in diesem System laufen, wie groß kann das Working-Set eines Prozesses maximal sein?

1 GiB RAM – (3 Prozesse * 16 MByte Seitentabelle)

– 128 MByte Betriebssystem = 848 MiB max. Working-Set Größe

5

Alle Gruppenmitglieder wissen gleich gut Bescheid

3

Aufgabe 4.5 (15):

InstantLab:

- LeakyApp WorkingSet wächst und wird dann wieder kleiner: Vergleich *Peak Mem Usage*, *Mem Usage*
2
- Effekt wird durch WorkingSet Trimming bewirkt, welcher auch bei anderen Prozessen beobachtet werden kann
2
- Pagefile Nutzung durch LeakyApp wächst VirtualMemory Size
- *Standby Liste*: Die enthaltenen Seiten gehörten zu einem Workingset. Die Daten sind (gegenüber einer Kopie im Pagefile) unverändert. Die entsprechenden Seiteneinträge verweisen weiterhin auf physikalische Seiten, sind aber mit „invalid“ und „transition“ gekennzeichnet.
2
- *Modified Liste*: Die enthaltenen Seiten gehörten zu einem Workingset. Die Daten wurden verändert und sind noch nicht auf die Platte/ins Pagefile geschrieben worden. Die entsprechenden Seiteneinträge verweisen weiterhin auf physikalische Seiten, sind aber mit „invalid“ und „transition“ gekennzeichnet. Die Seiten müssen vor der Wiederverwendung auf die Platte geschrieben werden.
2
- *Free Liste*: Seiten sind nicht mehr in Verwendung und enthalten unspezifizierte Daten. Aus Sicherheitsgründen müssen die Seiten vor der nächsten Verwendung erst mit ‚0‘ gefüllt werden.
2
- *Zero Liste*: Seiten sind frei und mit ‚0‘ gefüllt (vom Zero-page Thread)
2

Erklärung von soft- und hard-Page-Faults	2
Alle Gruppenmitglieder wissen gleich gut Bescheid	3