

Übungsblatt 5

Aufgabe 1 (Speicherverwaltung)

1. Kreuzen Sie an, bei welchen Konzepten der Speicherpartitionierung interne Fragmentierung entsteht.
☐ Statische Partitionierung
☐ Dynamische Partitionierung
☐ Buddy-Algorithmus
2. Kreuzen Sie an, bei welchen Konzepten der Speicherpartitionierung externe Fragmentierung entsteht.
☐ Statische Partitionierung
☐ Dynamische Partitionierung
☐ Buddy-Algorithmus
3. Geben Sie eine Möglichkeit an, um externe Fragmentierung zu beheben.
4. Kreuzen Sie an, welches Speicherverwaltungskonzept denjenigen freien Block sucht, der am besten passt.
☐ First Fit ☐ Next Fit ☐ Best fit ☐ Random
5. Kreuzen Sie an, welches Speicherverwaltungskonzept ab dem Anfang des Adressraums einen passenden freien Block sucht.
☐ First Fit ☐ Next Fit ☐ Best fit ☐ Random
6. Kreuzen Sie an, welches Speicherverwaltungskonzept den großen Bereich freien Speicher am Ende des Adressraums schnell zerstückelt.
☐ First Fit ☐ Next Fit ☐ Best fit ☐ Random
7. Kreuzen Sie an, welches Speicherverwaltungskonzept zufällig einen freien und passenden Block sucht.
☐ First Fit ☐ Next Fit ☐ Best fit ☐ Random
8. Kreuzen Sie an, welches Speicherverwaltungskonzept ab der Stelle der letzten Blockzuweisung einen passenden freien Block sucht.
☐ First Fit ☐ Next Fit ☐ Best fit ☐ Random
9. Kreuzen Sie an, welches Speicherverwaltungskonzept viele Minifragmente produziert und am langsamsten arbeitet.
☐ First Fit ☐ Next Fit ☐ Best fit ☐ Random

Aufgabe 2 (Buddy-Verfahren)

Das Buddy-Verfahren zur Zuweisung von Speicher an Prozesse soll für einen 1024 kB großen Speicher verwendet werden. Führen Sie die angegebenen Aktionen durch und geben Sie den Belegungszustand des Speichers nach jeder Anforderung oder Freigabe an.

	0	128	256	384	512	640	768	896	1024
Anfangszustand	1024 KB								
65 KB Anforderung => A									
30 KB Anforderung => B									
90 KB Anforderung => C									
34 KB Anforderung => D									
130 KB Anforderung => E									
Freigabe C									
Freigabe B									
275 KB Anforderung => F									
145 KB Anforderung => G									
Freigabe D									
Freigabe A									
Freigabe G									
Freigabe E									

Aufgabe 3 (Real Mode und Protected Mode)

1. Beschreiben Sie die Arbeitsweise des Real Mode.
2. Begründen Sie warum der Real Mode für Mehrprogrammbetrieb (Multitasking) ungeeignet ist.
3. Beschreiben Sie die Arbeitsweise des Protected Mode.
4. Beschreiben Sie was virtueller Speicher ist.
5. Erklären Sie, warum mit virtuellem Speicher der Hauptspeicher besser ausgenutzt wird.
6. Beschreiben Sie was Mapping ist.
7. Beschreiben Sie was Swapping ist.
8. Geben Sie den Namen der Komponente der CPU an, die virtuellen Speicher ermöglicht.
9. Beschreiben Sie die Aufgabe der Komponente aus Teilaufgabe 8.
10. Nennen Sie ein Konzept von virtuellem Speicher.

11. Geben Sie an, welche Form der Fragmentierung beim Konzept aus Teilaufgabe 10 entsteht.
12. Beschreiben Sie die Aufgabe und den Inhalt des Page-Table Base Register (PTBS).
13. Beschreiben Sie die Aufgabe und den Inhalt des Page-Table Length Register (PTLR).
14. Beschreiben Sie wie eine Page Fault Ausnahme (Exception) entsteht.
15. Beschreiben Sie wie das Betriebssystem auf eine Page Fault Ausnahme (Exception) reagiert.
16. Beschreiben Sie wie eine Access Violation Ausnahme (Exception) oder General Protection Fault Ausnahme (Exception) entsteht.
17. Beschreiben Sie die Auswirkung einer Access Violation Ausnahme (Exception) oder General Protection Fault Ausnahme (Exception).
18. Geben Sie an, was der Kernelspace enthält.
19. Geben Sie an, was der Userspace enthält.

Aufgabe 4 (Speicherverwaltung)

Kreuzen Sie bei jeder Aussage zur Speicherverwaltung an, ob die Aussage wahr oder falsch ist.

1. Real Mode ist für Multitasking-Systeme geeignet.
☐ Wahr ☐ Falsch
2. Beim Protected Mode läuft jeder Prozess in seiner eigenen, von anderen Prozessen abgeschotteten Kopie des physischen Adressraums.
☐ Wahr ☐ Falsch
3. Bei statischer Partitionierung entsteht interne Fragmentierung.
☐ Wahr ☐ Falsch
4. Bei dynamischer Partitionierung ist externe Fragmentierung unmöglich.
☐ Wahr ☐ Falsch
5. Beim Paging haben alle Seiten die gleiche Länge.
☐ Wahr ☐ Falsch

6. Ein Vorteil langer Seiten beim Paging ist geringe interne Fragmentierung.
- ☐ Wahr ☐ Falsch
7. Ein Nachteil kurzer Seiten beim Paging ist, dass die Seitentabelle sehr groß werden kann.
- ☐ Wahr ☐ Falsch
8. Die MMU übersetzt beim Paging logische Speicheradressen mit der Seitentabelle in physische Adressen.
- ☐ Wahr ☐ Falsch
9. Moderne Betriebssysteme (für x86) arbeiten im Protected Mode und verwenden ausschließlich Paging.
- ☐ Wahr ☐ Falsch

Aufgabe 5 (Seiten-Ersetzungsstrategien)

1. Erklären Sie, warum die optimale Ersetzungsstrategie OPT nicht implementiert werden kann.
2. Führen Sie die gegebene Zugriffsfolge mit den Ersetzungsstrategien Optimal, LRU, LFU und FIFO einmal mit einem Datencache mit einer Kapazität von 4 Seiten und einmal mit 5 Seiten durch. Berechnen Sie auch die Hitrate und die Missrate für alle Szenarien.

Optimale Ersetzungsstrategie (OPT):

Anfragen: **1 3 5 4 2 4 3 2 1 0 5 3 5 0 4 3 5 4 3 2 1 3 4 5**

Seite 1:																							
Seite 2:																							
Seite 3:																							
Seite 4:																							

Hitrate:

Missrate:

Anfragen: **1 3 5 4 2 4 3 2 1 0 5 3 5 0 4 3 5 4 3 2 1 3 4 5**

Seite 1:																							
Seite 2:																							
Seite 3:																							
Seite 4:																							
Seite 5:																							

Hitrate:

Missrate:

Ersetzungsstrategie Least Recently Used (LRU):

Anfragen: **1 3 5 4 2 4 3 2 1 0 5 3 5 0 4 3 5 4 3 2 1 3 4 5**

Seite 1:																							
Seite 2:																							
Seite 3:																							
Seite 4:																							

Hitrate:

Missrate:

Anfragen: **1 3 5 4 2 4 3 2 1 0 5 3 5 0 4 3 5 4 3 2 1 3 4 5**

Seite 1:																							
Seite 2:																							
Seite 3:																							
Seite 4:																							
Seite 5:																							

Hitrate:

Missrate:

Ersetzungsstrategie Least Frequently Used (LFU):

Anfragen: **1 3 5 4 2 4 3 2 1 0 5 3 5 0 4 3 5 4 3 2 1 3 4 5**

Seite 1:																							
Seite 2:																							
Seite 3:																							
Seite 4:																							

Queue:																							
--------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Hitrate:

Missrate:

Anfragen: **1 3 5 4 2 4 3 2 1 0 5 3 5 0 4 3 5 4 3 2 1 3 4 5**

Seite 1:																							
Seite 2:																							
Seite 3:																							
Seite 4:																							
Seite 5:																							

Queue:																							
--------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Hitrate:

Missrate:

Ersetzungsstrategie FIFO:

Anfragen: **1 3 5 4 2 4 3 2 1 0 5 3 5 0 4 3 5 4 3 2 1 3 4 5**

Seite 1:																							
Seite 2:																							
Seite 3:																							
Seite 4:																							

Hitrate:

Missrate:

Anfragen: **1 3 5 4 2 4 3 2 1 0 5 3 5 0 4 3 5 4 3 2 1 3 4 5**

Seite 1:																							
Seite 2:																							
Seite 3:																							
Seite 4:																							
Seite 5:																							

Hitrate:

Missrate:

3. Beschreiben Sie die Kernaussage der Anomalie von Laszlo Belady.
4. Zeigen Sie Belady's Anomalie, indem sie die gegebene Zugriffsfolge mit der Ersetzungsstrategie FIFO einmal mit einem Datencache mit einer Kapazität von 3 Seiten und einmal mit 4 Seiten durchführen. Berechnen Sie auch die Hitrate und die Missrate für beide Szenarien.

Anfragen: **3 2 1 0 3 2 4 3 2 1 0 4**

Seite 1:											
Seite 2:											
Seite 3:											

Hitrate:

Missrate:

Anfragen: **3 2 1 0 3 2 4 3 2 1 0 4**

Seite 1:											
Seite 2:											
Seite 3:											
Seite 4:											

Hitrate:

Missrate:

Aufgabe 6 (Zeitgesteuerte Kommandoausführung, Sortieren, Umgebungsvariablen)

1. Erzeugen Sie in Ihrem Benutzerverzeichnis (Home-Verzeichnis) ein Verzeichnis **Entbehrlich** und schreiben Sie einen Cron-Job, der immer Dienstags um 1:25 Uhr morgens den Inhalt von **Entbehrlich** löscht.

Die Ausgabe des Kommandos soll in eine Datei **LöschLog.txt** in Ihrem Home-Verzeichnis angehängt werden.

2. Schreiben Sie einen Cron-Job, der alle 3 Minuten zwischen 14:00 und 15:00 Uhr an jedem Dienstag im Monat November eine Zeile mit folgendem Aussehen (und den aktuellen Werten) an die Datei **Datum.txt** anhängt:

```
Heute ist der 30.10.2008
Die Uhrzeit ist 09:24:42 Uhr
*****
```

3. Schreiben Sie einen at-Job, der um 17:23 Uhr heute eine Liste der laufenden Prozesse ausgibt.

*Das Kommandozeilenwerkzeug **at** müssen Sie evtl. erst installieren.*
Unter Debian/Ubuntu geht das mit:
\$ sudo apt update && sudo apt install at
Unter CentOS/Fedora/RedHat geht das mit:
\$ sudo yum install at

4. Schreiben Sie einen **at**-Job, der am 24. Dezember um 8:15 Uhr morgens den Text „Endlich Weihnachten!“ ausgibt.
5. Erzeugen Sie in Ihrem Home-Verzeichnis eine Datei **Kanzler.txt** mit folgendem Inhalt:

Willy	Brandt	1969
Angela	Merkel	2005
Gerhard	Schröder	1998
KurtGeorg	Kiesinger	1966
Helmut	Kohl	1982
Konrad	Adenauer	1949
Helmut	Schmidt	1974
Ludwig	Erhard	1963
6. Geben Sie die Datei **Kanzler.txt** sortiert anhand der Vornamen aus.
7. Geben Sie die Datei **Kanzler.txt** sortiert anhand des dritten Buchstabens der Nachnamen aus.
8. Geben Sie die Datei **Kanzler.txt** sortiert anhand des Jahres der Amtseinführung aus.
9. Geben Sie die Datei **Kanzler.txt** rückwärts sortiert anhand des Jahres der Amtseinführung aus und leiten Sie die Ausgabe in eine Datei **Kanzlerdaten.txt**.
10. Erzeugen Sie mit dem Kommando **export** eine Umgebungsvariable **VAR1** und weisen Sie dieser den Wert **Testvariable** zu.
11. Geben Sie den Wert von **VAR1** in der Shell aus.
12. Löschen Sie die Umgebungsvariable **VAR1**.