Lösung von Übungsblatt 5

Aufgabe 1 (Speicherverwaltung)

| 1. | tierung? | | | | | | | | | |
|----|--|---|--------------------|----------------------------------|--|--|--|--|--|--|
| | ☑ Statische Par☑ Dynamische☑ Buddy-Algor | Partitionierung | | | | | | | | |
| 2. | 2. Bei welchen Konzepten der Speicherpartitionierung entsteht externe Fragmen tierung? | | | | | | | | | |
| | ☐ Statische Par☑ Dynamische☑ Buddy-Algor | Partitionierung | | | | | | | | |
| 3. | Wie kann exter | ne Fragmentieru | ing behoben we | rden? | | | | | | |
| | Durch Defragm rung keine Roll | _ | virtuellem Speic | her spielt externe Fragmentie- | | | | | | |
| 4. | Welches Konzep passt? | ot zur Speicherve | rwaltung sucht | den freien Block, der am besten | | | | | | |
| | \square First Fit | \square Next Fit | ⊠ Best fit | \square Random | | | | | | |
| 5. | | pt zur Speicherv ssenden freien B | _ | at ab dem Anfang des Adress- | | | | | | |
| | ⊠ First Fit | \square Next Fit | \square Best fit | \square Random | | | | | | |
| 6. | _ | pt zur Speicherv icher am Ende d | _ | tückelt schnell den großen Bes? | | | | | | |
| | \square First Fit | \boxtimes Next Fit | \square Best fit | \square Random | | | | | | |
| 7. | Welches Konzepsenden Block? | pt zur Speicherv | erwaltung wähl | t zufällig einen freien und pas- | | | | | | |
| | \square First Fit | \square Next Fit | \square Best fit | \boxtimes Random | | | | | | |
| 8. | _ | ot zur Speicherven n passenden freie | _ | ab der Stelle der letzten Block- | | | | | | |
| | \square First Fit | ⊠ Next Fit | \square Best fit | ☐ Random | | | | | | |
| | | | | | | | | | | |

Inhalt: Themen aus Foliensatz 5

| 9. | Welches | Konzep | ot zur Speic | cherve | rwaltung | produz | iert vie | le Minifi | ragmente | und |
|----|-----------|--------|--------------|--------|----------|--------|----------|-----------|----------|-----|
| | arbeit et | am lan | gsamsten? | | | | | | | |
| | ☐ First | Fit | □ Nevt F | it | ⊠ Rest f | it | Ranc | dom | | |

Aufgabe 2 (Buddy-Verfahren)

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

| | 1024 KB | | | | | | | | | |
|--|---------|----------------|----------|--------|---------------|--------|--|--|--|--|
| 65 KB Anforderung => A | А | 128 KB | 256 | КВ | 512 KB | | | | | |
| 30 KB Anforderung => B | А | B 32 64 KB | 256 | КВ | 512 KB | | | | | |
| 90 KB Anforderung => C | А | B 32 64 KB | С | 128 KB | 512 KB | | | | | |
| 34 KB Anforderung => D | А | B 32 D | C 128 KB | | 512 KB | | | | | |
| 130 KB Anforderung => E | А | B 32 D | С | 128 KB | Е | 256 KB | | | | |
| Freigabe C | А | B 32 D | 128 KB | 128 KB | Е | 256 KB | | | | |
| | А | B 32 D | 256 | КВ | Е | 256 KB | | | | |
| Freigabe B | А | 32 32 D | 256 | КВ | E | 256 KB | | | | |
| | А | 64 KB D | 256 | KB | E | 256 KB | | | | |
| 275 KB Anforderung => F Nicht möglich, weil keine 275 kB am Stück frei | А | 64 KB D | 256 | КВ | Е | 256 KB | | | | |
| 145 KB Anforderung => G | А | 64 KB D | G | | Е | 256 KB | | | | |
| Freigabe D | А | 64 KB 64 KB | (| 3 | Е | 256 KB | | | | |
| | А | 128 KB | (| G . | Е | 256 KB | | | | |
| Freigabe A | 128 KB | 128 KB | (| 3 | E | 256 KB | | | | |
| | 256 | S KB | (| ĵ | E | 256 KB | | | | |
| Freigabe G | 128 KB | 128 KB | 256 | КВ | Е | 256 KB | | | | |
| | | 512 | КВ | | Е | 256 KB | | | | |
| Freigabe E | | 512 | КВ | | 256 KB 256 KB | | | | | |
| | | 512 | КВ | | 512 KB | | | | | |
| | 1024 KB | | | | | | | | | |

Inhalt: Themen aus Foliensatz 5 Seite 2 von 11

Aufgabe 3 (Real Mode und Protected Mode)

1. Wie arbeitet der Real Mode?

Jeder Prozess kann direkt auf den gesamten adressierbaren Speicher zugreifen.

2. Warum ist der Real Mode für Mehrprogrammbetrieb (Multitasking) ungeeignet?

Es gibt keinen Speicherschutz.

3. Wie arbeitet der Protected Mode?

Jeder Prozess darf nur auf seinen eigenen virtuellen Speicher zugreifen. Virtuelle Speicheradressen übersetzt die CPU mit Hilfe der MMU in physische Speicheradressen.

4. Was ist virtueller Speicher?

Jeder Prozess besitzt einen eigenen Adressraum. Der Adressraum ist eine Abstraktion des physischen Speichers. Es handelt sich dabei um virtuellen Speicher. Er besteht aus logischen Speicheradressen, die von der Adresse 0 aufwärts durchnummeriert sind und er ist unabhängig von der verwendeten Speichertechnologie und den gegebenen Ausbaumöglichkeiten.

5. Erklären Sie, warum mit virtuellem Speicher der Hauptspeicher besser ausgenutzt wird.

Die Prozesse müssen nicht am Stück im Hauptspeicher liegen. Externe Fragmentierung entsteht, spielt aber keine Rolle.

6. Was ist Mapping?

Abbilden des virtuellen Speichers auf den realen Speicher.

7. Was ist Swapping?

Prozess des Ein- und Auslagerns von Daten in den/vom Arbeitsspeicher vom/in den Hintergrundspeicher (Festplatten/SSDs).

8. Welche Komponente der CPU ermöglicht virtuellen Speicher?

Memory Management Unit (MMU).

9. Was genau ist die Aufgabe der Komponente aus Teilaufgabe 8?

Virtuelle Speicheradressen übersetzt die CPU mit Hilfe der MMU in physische Speicheradressen.

Inhalt: Themen aus Foliensatz 5 Seite 3 von 11

10. Nennen Sie ein Konzept von virtuellem Speicher.

Paging.

11. Welche Form der Fragmentierung entsteht bei dem Konzept aus Teilaufgabe 10?

Interne Fragmentierung entsteht beim Paging (aber nur in der letzten Seite eines Prozesses).

12. Wie entsteht eine Page Fault Ausnahme (Exception)?

Ein Prozess versucht auf eine Seite zuzugreifen, die nicht im physischen Hauptspeicher ist.

13. Wie reagiert das Betriebssystem auf eine Page Fault Ausnahme (Exception)?

Das Betriebssystem behandelt die Ausnahme mit folgenden Schritten:

- Daten auf dem Sekundärspeicher (SDD/HDD) lokalisieren.
- Die Seite in eine freie Hauptspeicherseite laden.
- Seitentabelle aktualisieren.
- Kontrolle an den Prozess zurückgeben. Dieses fährt die Anweisung, die zum Page Fault führte, erneut aus.
- 14. Wie entsteht eine Access Violation Ausnahme (Exception) oder General Protection Fault Ausnahme (Exception)?

Ein Prozess versucht auf eine virtuelle Speicheradresse zuzugreifen, auf die er nicht zugreifen darf.

15. Welche Auswirkung hat eine Access Violation Ausnahme (Exception) oder General Protection Fault Ausnahme (Exception)?

Bei einigen Windows-Betriebssystemen aus der Vergangenheit waren Schutzverletzungen häufig ein Grund für Systemabstürze und hatten einen "Blue Screen" zur Folge. Unter Linux wird als Ergebnis das Signal SIGSEGV erzeugt.

16. Was enthält der Kernelspace?

Den Betriebssystemkern (Kernel) und Kernelerweiterungen (Treiber).

17. Was enthält der Userspace?

Den aktuell ausgeführten Prozess, der um den Erweiterungsspeicher ("Swap", Windows: "Page-File") vergrößert wird.

Inhalt: Themen aus Foliensatz 5

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 | \square Falsch | | | | | | | |
| 3. | Bei statischer I | Partitionierung entsteht interne Fragmentierung. | | | | | | | |
| | ⊠ Wahr | \square Falsch | | | | | | | |
| 4. | Bei dynamische | er Partitionierung ist externe Fragmentierung unmöglich. | | | | | | | |
| | \square Wahr | ⊠ Falsch | | | | | | | |
| 5. | Beim Paging h | aben alle Seiten die gleiche Länge. | | | | | | | |
| | ⊠ Wahr | \square Falsch | | | | | | | |
| 6. | Ein Vorteil lang | ger Seiten beim Paging ist geringe interne Fragmentierung. | | | | | | | |
| | \square Wahr | ⊠ Falsch | | | | | | | |
| 7. | Ein Nachteil k werden kann. | urzer Seiten beim Paging ist, das die Seitentabelle sehr groß | | | | | | | |
| | ⊠ Wahr | \square Falsch | | | | | | | |
| 8. | Die MMU übersetzt beim Paging logische Speicheradressen mit der Seitentabelle in physische Adressen. | | | | | | | | |
| | ⊠ Wahr | \square Falsch | | | | | | | |
| 9. | Moderne Betrie den ausschließl | ebssysteme (für x86) arbeiten im Protected Mode und verwenich Paging. | | | | | | | |
| | ⊠ Wahr | ☐ Falsch | | | | | | | |
| | | | | | | | | | |
| 11 | fgabe 5 | (Seiten-Ersetzungsstrategien) | | | | | | | |

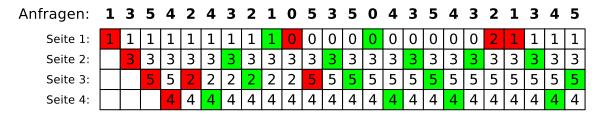
Aufgabe 5 (Seiten-Ersetzungsstrategien)

Weil man nicht in die Zukunft schauen kann und damit ist die zukünftige Zugriffsfolge unbekannt.

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):

Hinweis: Wenn bei der optimalen Ersetzungsstrategie eine Seite verdrängt werden muss, wird die Seite verdrängt, auf die am längsten in der Zukunft nicht zugegriffen wird.

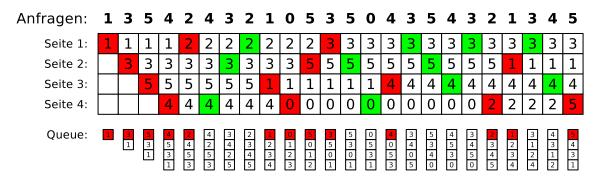


Hitrate: 15/24 = 0,625Missrate: 9/24 = 0,375

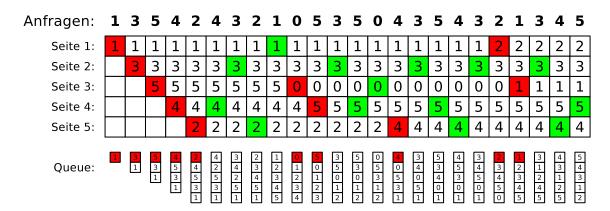
| Anfragen: | 1 | 3 | 5 | 4 | 2 | 4 | 3 | 2 | 1 | 0 | 5 | 3 | 5 | 0 | 4 | 3 | 5 | 4 | 3 | 2 | 1 | 3 | 4 | 5 |
|-----------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Seite 1: | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Seite 2: | | 3 | 3 | 3 | 3 | Э | 3 | 3 | 3 | 3 | З | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| Seite 3: | | | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| Seite 4: | | | | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| Seite 5: | | | | | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 |

Hitrate: 17/24 = 0,7083333Missrate: 7/24 = 0,2916666 Ersetzungsstrategie Least Recently Used (LRU):

Hinweis: Wenn bei der Ersetzungsstrategie LRU eine Seite verdrängt werden muss, wird die Seite verdrängt, auf die am längsten nicht zugegriffen wurde.



Hitrate: 11/24 = 0,4583333%Missrate: 13/24 = 0,5416666%

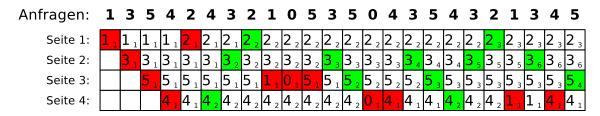


Hitrate: 14/24 = 0.583333%Missrate: 10/24 = 0.416666%

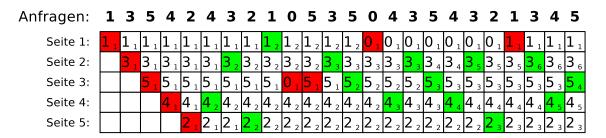
Ersetzungsstrategie Least Frequently Used (LFU):

Hinweis: Wenn bei der Ersetzungsstrategie LFU eine Seite verdrängt werden muss, wird die Seite verdrängt, auf die am wenigsten zugegriffen wurde. Es wird für jede Seite in der Seitentabelle ein Referenzzähler geführt, der die

Anzahl der Zugriffe speichert. Ist der Speicher voll und kommt es zum Miss, wird die Seite entfernt, deren Referenzzähler den niedrigsten Wert hat.



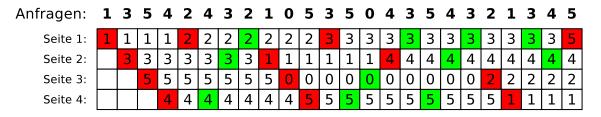
Hitrate: 12/24 = 0.5Missrate: 12/24 = 0.5



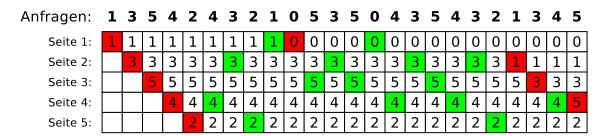
Hitrate: 15/24 = 0,625Missrate: 9/24 = 0,375

Ersetzungsstrategie FIFO:

Hinweis: Wenn bei der Ersetzungsstrategie FIFO eine Seite verdrängt werden muss, wird die Seite verdrängt, die sich am längsten im Speicher befindet.



Hitrate: 11/24 = 0,4583333Missrate: 13/24 = 0,5416666



Hitrate: 15/24 = 0,625Missrate: 9/24 = 0,375 3. Was ist die Kernaussage der Anomalie von Laszlo Belady?

FIFO führt bei bestimmten Zugriffsmustern bei einem vergrößerten Speicher zu schlechteren Ergebnissen.

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: | 3 | 3 | 3 | 0 | 0 | 0 | 4 | 4 | 4 | 4 | 4 | 4 |
| Seite 2: | | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 | 1 | 1 | 1 |
| Seite 3: | | | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 0 | 0 |

Hitrate: 3/12 = 25%Missrate: 9/12 = 75%

| Anfragen: | 3 | 2 | 1 | 0 | 3 | 2 | 4 | 3 | 2 | 1 | 0 | 4 |
|-----------|---|---|---|---|---|---|---|---|---|---|---|---|
| Seite 1: | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 4 | 0 | 0 |
| Seite 2: | | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | ო | Ж | 4 |
| Seite 3: | | | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| Seite 4: | | | | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |

Hitrate: 2/12 = 16,66%Missrate: 10/12 = 83,33%

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.

\$ mkdir ~/Entbehrlich

\$ crontab -e

Folgende Zeile eintragen:

25 1 * * 2 rm -rfv /home/USERNAME/Entbehrlich/* >> /home/USERNAME/LöschLog.txt

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

\$ crontab -e

Folgende Zeile eintragen:

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

\$ at 1725 today

Folgende Zeile eintragen:

ps -r

4. Schreiben Sie einen at-Job, der am 24. Dezember um 8:15 Uhr morgens den Text "Endlich Weihnachten!" ausgibt.

\$ at 0815 DEZ 25

Folgende Zeile eintragen:

echo "Endlich Weihnachten!"

5. Erzeugen Sie in Ihrem Home-Verzeichnis eine Datei Kanzler.txt mit folgendem Inhalt:

Willy Brandt 1969

Merkel

Angela

| | 111160±a | 116014 11011101 | | | | |
|-------------------|----------------|-----------------|-----------|-------|----|---------------|
| Gerhard Schröder | | | 1998 | | | |
| KurtGeorg Kiesing | | org Kiesinge | er 1966 | | | |
| | Helmut Kohl | | 1982 | | | |
| | ${\tt Konrad}$ | Adenauer | 1949 | | | |
| | ${\tt Helmut}$ | Helmut Schmidt | | | | |
| | Ludwig | Ludwig Erhard | | | | |
| | | | | | | |
| | \$ echo | "Willy | Brandt | 1969" | >> | ~/Kanzler.txt |
| | \$ echo | "Angela | Merkel | 2005" | >> | ~/Kanzler.txt |
| | \$ echo | "Gerhard | Schröder | 1998" | >> | ~/Kanzler.txt |
| | \$ echo | "KurtGeorg | Kiesinger | 1966" | >> | ~/Kanzler.txt |
| | \$ echo | "Helmut | Kohl | 1982" | >> | ~/Kanzler.txt |
| | \$ echo | "Konrad | Adenauer | 1949" | >> | ~/Kanzler.txt |
| | \$ echo | "Helmut | Schmidt | 1974" | >> | ~/Kanzler.txt |
| | \$ echo | "Ludwig | Erhard | 1963" | >> | ~/Kanzler.txt |
| | | | | | | |

2005

- 6. Geben Sie die Datei Kanzler.txt sortiert anhand der Vornamen aus.
 - \$ sort ~/Kanzler.txt
- 7. Geben Sie die Datei Kanzler.txt sortiert anhand des dritten Buchstabens der Nachnamen aus.
 - \$ sort -k+2.4 ~/Kanzler.txt
- 8. Geben Sie die Datei Kanzler.txt sortiert anhand des Jahres der Amtseinführung aus.
 - \$ sort -k3 ~/Kanzler.txt
- 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.
 - \$ sort -k3 -nr ~/Kanzler.txt > ~/Kanzlerdaten.txt
- 10. Erzeugen Sie mit dem Kommando export eine Umgebungsvariable VAR1 und weisen Sie dieser den Wert Testvariable zu.
 - \$ export VAR01=Testvariable
- 11. Geben Sie den Wert von VAR1 in der Shell aus.
 - \$ printenv VAR01
- 12. Löschen Sie die Umgebungsvariable VAR1.
 - \$ unset VAR01