

Lösung von Übungsblatt 4

Aufgabe 1 (Festplatten)

1. Beschreiben Sie, was bei man bei Festplatten unter Sektoren (= Blöcken) versteht.

Die Spuren sind in kleine logische Einheiten (Kreissegmente) unterteilt, die Blöcke oder Sektoren heißen. Sektoren sind die kleinsten adressierbaren Einheiten auf Festplatten.

2. Beschreiben Sie, was bei man bei Festplatten unter Spuren versteht.

Die Oberflächen der Scheiben bei Festplatten werden in kreisförmigen Spuren („Tracks“) von den Köpfen magnetisiert.

3. Beschreiben Sie, was bei man bei Festplatten unter Zylindern versteht.

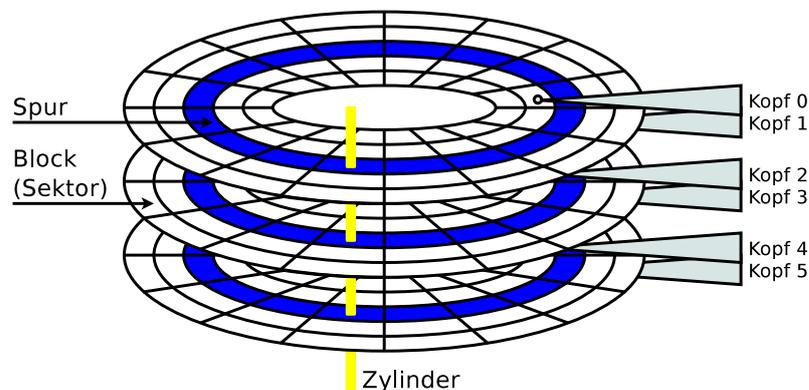
Alle Spuren auf allen Platten bei einer Position des Schwingarms bilden einen Zylinder („Cylinder“)

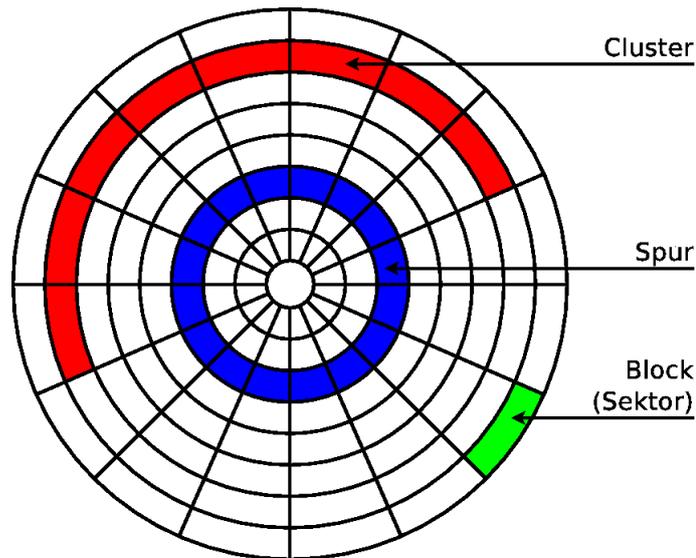
4. Beschreiben Sie, was bei man bei Festplatten unter Clustern versteht.

Cluster sind Verbände von Sektoren mit fester Größe, z.B. 4 oder 8 kB. Bei modernen Betriebssystemen sind Cluster die kleinste Zuordnungseinheit.

5. Zeichnen Sie den Aufbau einer Festplatte schematisch. Machen Sie anhand Ihrer Zeichnung(en) deutlich, was folgende Begriffe bedeuten:

- a) Sektor (= Block)
- b) Spur
- c) Zylinder
- d) Cluster





6. Begründen Sie, warum die Geschwindigkeit (insbesondere die Zugriffszeit) bei Festplatten nicht beliebig gesteigert werden kann.

Festplatten bestehen aus beweglichen Teilen. Physikalische und materielle Grenzen müssen akzeptiert werden.

7. Geben Sie die Faktoren an, die die Zugriffszeit einer Festplatte beeinflussen.
- Suchzeit („Average Seek Time“)
 - Zugriffsverzögerung durch Umdrehung („Average Rotational Latency Time“)
8. Beschreiben Sie die Faktoren aus Teilaufgabe 7.
- Suchzeit („Average Seek Time“): Die Zeit, die der Schwungarm braucht, um eine Spur zu erreichen.
 - Zugriffsverzögerung durch Umdrehung („Average Rotational Latency Time“): Verzögerung der Drehgeschwindigkeit bis der Schreib-/Lesekopf den gewünschten Block erreicht. Hängt ausschließlich von der Drehgeschwindigkeit der Scheiben ab.

Aufgabe 2 (Festplattengeometrie)

Auf einer älteren Festplatte befinden sich folgende Informationen:

```
Western Digital WD Caviar 64AA          Enhanced IDE Hard Drive
Drive parameters    13328 cyl    15 heads    63 spt    6448.6 MB
S/N: WM653  321 5163    MDL: WD64AA - 00AAA4    DATE: 02 FEB 2000
```

1. Berechnen Sie die Kapazität einer Scheibe der Festplatte.
(Bei der Lösung muss der Rechenweg angegeben sein!)

Die Anzahl der Zylinder (*cy1*) ist identisch mit der Anzahl der Spuren (= Tracks) pro Scheibe.

Lösung: Die Kapazität einer Oberfläche einer Scheibe = Anzahl der Zylinder * Anzahl der Sektoren pro Spur bzw. Track (*spt*) * Bytes pro Sektor bzw. Block. Die Größe der Sektoren ist 512 Byte.

$$13.328 * 63 * 512 = 429.907.968 \text{ Byte}$$

Jede Scheibe hat zwei Oberflächen. Das heißt, dass die beiden Oberflächen einer Scheibe zusammen eine Kapazität von 859.815.936 Byte haben.

Bei 15 Köpfen werden bei 7 Scheiben beide Oberflächen verwendet und bei einer Scheibe wird nur eine Oberfläche verwendet.

2. Berechnen Sie die Größe einer Spur der Festplatte.
(Bei der Lösung muss der Rechenweg angegeben sein!)

Lösung: Größe einer Spur = Anzahl der Sektoren pro Spur bzw. Track (*spt*) * Bytes pro Sektor bzw. Block.

$$63 * 512 = 32.256 \text{ Byte}$$

3. Berechnen Sie die Gesamtkapazität der Festplatte.
(Bei der Lösung muss der Rechenweg angegeben sein!)

Die Anzahl der Köpfe (*heads*) ist identisch mit der Anzahl der Oberflächen der Scheiben.

Lösung: Gesamtkapazität = Kapazität einer Oberfläche einer Scheibe * Anzahl der Köpfe (*heads*).

$$429.907.968 * 15 = 6.448.619.520 \text{ Byte}$$

4. Geben Sie an, ob die Angaben auf der Festplatte der physischen Geometrie entsprechen
(Begründen Sie Ihre Antwort!)

Es macht keinen Sinn, dass eine Festplatte 15 Köpfe hat. Bei diesen Werten wird eine andere (logische) Geometrie (Zylinder/Köpfe/Sektor) verwendet, als die physische Geometrie. Diese Angaben und die damit verbundene logische Geometrie, die mit der physischen Geometrie nichts zu tun hat, ergeben sich aus den Einschränkungen des BIOS.

Aufgabe 3 (Solid State Drives)

1. Begründen Sie, warum es falsch ist, SSDs als Solid State Disks zu bezeichnen.

SSDs enthalten keine beweglichen Teile und darum auch keine „Disks“.

2. Nennen Sie vier Vorteile von SSDs gegenüber Festplatten.

- *Kurze Zugriffszeit*
- *Geringer Energieverbrauch*
- *Keine Geräuschentwicklung*
- *Mechanische Robustheit*
- *Geringes Gewicht*
- *Die Position der Daten ist irrelevant \implies Defragmentieren ist sinnlos*

3. Nennen Sie zwei Nachteile von SSDs gegenüber Festplatten.

- *Höherer Preis im Vergleich zu Festplatten gleicher Kapazität*
- *Sicheres Löschen bzw. Überschreiben ist schwierig*
- *Eingeschränkte Anzahl an Schreib-/Löschzyklen*

4. Begründen Sie, warum Löschvorgänge bei Flash-Speicher aufwendiger als Leseoperationen sind.

Die Speicherzellen sind in Gruppen zu Seiten und Blöcken angeordnet. Je nach dem Aufbau eines Flash-Speichers, enthält ein Block immer eine feste Anzahl an Seiten. Schreib- und Löschoptionen können nur für komplette Seiten oder Blöcke durchgeführt werden.

5. Nennen Sie einen Vorteil und einen Nachteil von NOR-Speicher.

- *Vorteil:*
 - *Wahlfreier Lese- und Schreibzugriff \implies Bessere Zugriffszeit als NAND-Speicher*
- *Nachteil:*
 - *Komplexer (\implies kostspieliger) Aufbau als NAND-Speicher*
 - *Höherer Stromverbrauch als NAND-Speicher*
 - *Üblicherweise geringe Kapazitäten (≤ 32 MB)*

6. Nennen Sie einen Vorteil und einen Nachteil von NAND-Speicher.

- *Vorteil:*
 - *Weniger Datenleitungen \implies Benötigt weniger Fläche als NOR-Speicher*
 - *Herstellung ist preisgünstiger als NOR-Flash-Speicher*
- *Nachteil:*
 - *Kein wahlfreier Zugriff \implies Schlechtere Zugriffszeit als NOR-Speicher*
 - *Lese- und Schreibzugriffe sind nur für ganze Seiten möglich*
 - *Löschoptionen sind nur für ganze Blöcke möglich*

7. Beschreiben Sie den Unterschied zwischen NAND-Speicher der Kategorien Single-Level Cell (SLC), Multi-Level Cell (MLC) und Triple-Level Cell (TLC).
 - *TLC-Speicherzellen speichern 3 Bits*
 - *MLC-Speicherzellen speichern 2 Bits*
 - *SLC-Speicherzellen speichern 1 Bit*
8. Beschreiben Sie die Aufgabe eines Wear Leveling-Algorithmus bei Flash-Speicher.

Wear Leveling-Algorithmen verteilen Schreibzugriffe gleichmäßig.

Aufgabe 4 (RAID)

1. Geben Sie an, welche RAID-Level die Datentransferrate beim Schreiben verbessern.

RAID-0 RAID-1 RAID-5

2. Geben Sie an, welche RAID-Level die Ausfallsicherheit verbessern.

RAID-0 RAID-1 RAID-5

3. Geben Sie an, wie viele Laufwerke bei einem RAID-0-Verbund ausfallen dürfen, ohne dass es zum Datenverlust kommt.

Es dürfen keine Laufwerke ausfallen.

4. Geben Sie an, wie viele Laufwerke bei einem RAID-1-Verbund ausfallen dürfen, ohne dass es zum Datenverlust kommt.

Es muss mindestens noch ein Laufwerk funktionieren.

5. Geben Sie an, wie viele Laufwerke bei einem RAID-5-Verbund ausfallen dürfen, ohne dass es zum Datenverlust kommt.

Es darf maximal ein Laufwerk ausfallen.

6. Nehmen Sie Stellung zu der Aussage: „Ein RAID-Verbund kann das regelmäßige Backup wichtiger Daten ersetzen“.

RAID ist kein Ersatz für Datensicherung. Fehlerhafte Dateioperationen oder Virenbefall finden auf allen Laufwerken statt.

7. Begründen Sie, warum es nicht sinnvoll ist, Paritätsinformationen auf nur einem Laufwerk zu speichern, sondern auf allen Laufwerken zu verteilen.

Jede Schreiboperation auf das RAID führt zu Schreiboperationen auf das Paritätslaufwerk \implies Flaschenhals.

8. Geben Sie die Nettokapazität eines RAID-0-Verbunds an.

Die Nettokapazität ist n , wobei n die Anzahl der Laufwerke ist.

9. Geben Sie die Nettokapazität eines RAID-1-Verbunds an.

Die Nettokapazität entspricht der Kapazität des kleinsten Laufwerks.

10. Geben Sie die Nettokapazität eines RAID-5-Verbunds an.

Die Nettokapazität ist $n - 1$, wobei n die Anzahl der Laufwerke ist.

11. Beschreiben Sie wie die Berechnung der Paritätsinformationen bei RAID-5 funktioniert.

$P(16-19) = \text{Block 16 XOR Block 17 XOR Block 18 XOR Block 19}$.

12. Nennen Sie einen Vorteil und einen Nachteil von Software-RAID gegenüber Hardware-RAID.

Ein Vorteil ist zum Beispiel: Keine Kosten für zusätzliche Hardware.

Nachteile sind zum Beispiel die Betriebssystemabhängigkeit und zusätzliche CPU-Belastung bei Lese- und Schreiboperationen.

Aufgabe 5 (Zeichen zählen, Zeit und Datum, Aliase, Weiterleitung, Dateien suchen)

1. Erstellen Sie mit dem Kommando `echo` eine Datei `Zitat.txt` mit folgendem Inhalt:

```
Was man nicht weiß,  
das eben brauchte man,  
und was man weiß,  
kann man nicht brauchen.
```

Gothe (Faust)

```
$ echo -e "Was man nicht weiß,\ndas eben brauchte man,\nund was man weiß,\nkann man nicht brauchen.\n\nGothe (Faust)" > Zitat.txt
```

2. Lassen Sie sich mit dem Kommando `wc` die Anzahl der Zeichen in der Datei `Zitat.txt` ausgeben.

```
$ wc -m Zitat.txt
```

3. Lassen Sie sich die Anzahl der Worte in der Datei `Zitat.txt` ausgeben, indem Sie den Inhalt der Datei in der Shell ausgeben und in die Eingabe von `wc` leiten.

```
$ wc -w Zitat.txt
```

4. Lassen Sie sich den Kalender des Jahres 1999 ausgeben und leiten Sie die Ausgabe in eine neue Datei `Kalender.txt`.

```
$ cal 1999 > Kalender.txt
```

5. Erzeugen Sie dem Kommando `date` in der Shell eine Ausgabe mit Informationen zum aktuellen Datum in der dargestellten Form:

```
Heute ist Donnerstag, der 24. Oktober 2013.
```

```
Es ist 16:08 Uhr und 07 Sekunden.
```

```
In UNIX-Zeit ist es genau: 1382623687
```

Hängen Sie die Ausgabe durch Weiterleitung an die Datei `Kalender.txt` an.

```
$ date +"Heute ist %A, der %d. %B %Y.%nEs ist %H:%M Uhr und %S Sekunden.%nIn UNIX-Zeit ist es genau: %s" >> Kalender.txt
```

6. Lassen Sie die Anzahl der Einträge (Dateien und Verzeichnisse) im Verzeichnis `/dev` mit `wc` berechnen. Dabei soll auch die Abarbeitungsgeschwindigkeit gemessen werden.

```
$ time ls -l /dev | wc -l
```

7. Lassen Sie eine Liste der existierenden Aliase ausgeben.

```
$ alias
```

8. Legen Sie ein Alias `zeit` an, das die in Teilaufgabe 6 gesuchte Ausgabe erzeugt.

```
$ alias zeit="time ls -l /dev | wc -l"
```

9. Entfernen Sie das Alias `zeit`.

```
$ unalias zeit
```

10. Suchen Sie mit einem Kommando in ihrem Home-Verzeichnis alle Dateien, auf die folgende Kriterien passen:

- Es sollen nur Dateien (keine Verzeichnisse oder Links) gefunden werden.
- Der Dateiname muss den String `BTS` (Groß-/Kleinschreibung ignorieren) enthalten.
- Die Dateien sollen Ihnen (User-ID) gehören.
- Das Alter der Dateien soll mindestens 1 Tag sein.
- Die letzte Änderung soll vor mehr als 3 Tagen stattgefunden haben.
- Die Dateigröße soll mindestens 10 Kilobyte betragen.

Gleichzeitig soll zu jeder gefunden Datei die Anzahl der enthaltenen Zeilen ausgegeben werden.

```
$ find . -iname "*bts*" -user USERNAME -atime -1 -mtime +3  
-type f -size +10k -exec wc -l "{}" ";"
```