

Übungsblatt 3

Aufgabe 1 (Dateisysteme)

1. Nennen Sie die Informationen, die ein Inode speichert.
2. Nennen Sie drei Beispiele für Metadaten im Dateisystem.
3. Beschreiben Sie was ein Cluster im Dateisystem ist.
4. Beschreiben Sie wie ein UNIX-Dateisystem (z.B. ext2/3), das keine Extents verwendet, mehr als 12 Cluster adressieren kann.
5. Beschreiben Sie wie Verzeichnisse bei Linux-Dateisystemen technisch realisiert sind.
6. Nennen Sie einen Vorteil und einen Nachteil kleiner Cluster im Dateisystem im Gegensatz zu großen Clustern.
7. DOS/Windows-Dateisysteme unterscheiden Groß- und Kleinschreibung.
☐ Wahr ☐ Falsch
8. UNIX-Dateisysteme unterscheiden Groß- und Kleinschreibung.
☐ Wahr ☐ Falsch
9. Moderne Betriebssysteme beschleunigen Zugriffe auf gespeicherte Daten mit einem Cache im Hauptspeicher.
☐ Wahr ☐ Falsch
10. Die meisten Betriebssysteme arbeiten nach dem Prinzip...
☐ Write-Back ☐ Write-Through
11. Nennen Sie je einen Vorteil und einen Nachteil eines Caches im Hauptspeicher, mit dem Betriebssysteme die Zugriffe auf gespeicherte Daten beschleunigen.
12. Beschreiben Sie was ein absoluter Pfadname ist.
13. Beschreiben Sie was ein relativer Pfadname ist.
14. `/var/log/messages` ist ein...
☐ Absoluter Pfadname ☐ Relativer Pfadname
15. `BTS_Vorlesung/Vorlesung_05/folien_bts_vorlesung_05.tex` ist ein...
☐ Absoluter Pfadname ☐ Relativer Pfadname

16. `Dokumente/MasterThesis/thesis.tex` ist ein...

- ☐ Absoluter Pfadname ☐ Relativer Pfadname

17. `/home/<benutzername>/Mail/inbox/` ist ein...

- ☐ Absoluter Pfadname ☐ Relativer Pfadname

18. Nennen Sie die Informationen, die der Bootsektor eines Dateisystems speichert.

19. Nennen Sie die Informationen, die der Superblock eines Dateisystems speichert.

20. Beschreiben Sie, warum manche Dateisysteme (z.B. ext2/3) die Cluster des Dateisystems zu Blockgruppen zusammenfassen.

21. Beschreiben Sie, was die Dateizuordnungstabelle bzw. File Allocation Table (FAT) ist, und nennen Sie die Informationen, die diese enthält.

22. Beschreiben Sie die Aufgabe des Journals bei Journaling-Dateisystemen.

23. Nennen Sie einen Vorteil von Journaling-Dateisystemen gegenüber Dateisystemen ohne Journal.

24. Beschreiben Sie den Vorteil von Extents gegenüber direkter Adressierung der Cluster.

25. Stellen Sie sich ein Dateisystem mit einem unendlich großen (oder zumindest sehr großen) Blockspeichergerät vor. Nennen und beschreiben Sie einen begrenzenden Faktor, der Sie daran hindert, eine unendliche Anzahl von Dateien zu erstellen. (*Die Speicherkapazität des Blockspeichers ist hier nicht der begrenzende Faktor!*)

26. Einige Dateisysteme verwenden ein Konzept namens Copy-on-write (COW). Kreuzen Sie die beiden Antworten an, die auf solche Dateisysteme zutreffen.

Wenn eine Datei geändert wird, werden die alten Cluster im Dateisystem, die geändert werden müssen,...

- ☐ beibehalten (nicht geändert).
☐ mit den neuen Änderungen überschrieben.
☐ gelöscht, indem die Clusteradresse im Inode entfernt wird.
☐ in neue Cluster kopiert, in denen die Änderungen vorgenommen werden.

Aufgabe 2 (Dateisysteme)

Kreuzen Sie bei jeder Aussage zu Dateisystemen an, ob die Aussage wahr oder falsch ist.

Aussage	wahr	falsch
Inodes speichern alle Verwaltungsdaten (Metadaten) der Dateien.		
Dateisysteme adressieren Cluster und nicht Blöcke des Datenträgers oder Laufwerks.		
Je kleiner die Cluster, desto größer ist der Verwaltungsaufwand für große Dateien.		
Je größer die Cluster, desto geringer ist der Kapazitätsverlust durch interne Fragmentierung.		
Unter UNIX haben Dateiendungen schon immer eine große Bedeutung.		
Moderne Dateisysteme arbeiten so effizient, dass Puffer durch das Betriebssystem nicht mehr üblich sind.		
Absolute Pfadnamen beschreiben den kompletten Pfad von der Wurzel bis zur Datei.		
Das Trennzeichen in Pfadangaben ist bei allen Betriebssystemen gleich.		
Ein Vorteil der Blockgruppen bei ext2 ist, dass die Inodes physisch nahe bei den Clustern liegen, die sie adressieren.		
Eine Dateizuordnungstabelle (FAT) erfasst die belegten und freien Cluster im Dateisystem.		
Bei der Master File Table von NTFS ist Fragmentierung unmöglich.		
Ein Journal im Dateisystem reduziert die Anzahl der Schreibzugriffe.		
Journaling-Dateisysteme grenzen die bei der Konsistenzprüfung zu überprüfenden Daten ein.		
Bei Dateisystemen mit Journal sind Datenverluste garantiert ausgeschlossen.		
Vollständiges Journaling führt alle Schreiboperation doppelt aus.		
Extents verursachen weniger Verwaltungsaufwand als Blockadressierung.		

Aufgabe 3 (Dateisysteme)

Gegeben ist folgender Dateisystembaum:

```
/
├── bin
├── boot
├── dev
├── ...
├── img
│   ├── logo
│   └── thumb
├── src                                <--- (2)
│   ├── factories                      <--- (3)
│   ├── adapters
│   ├── main
│   │   ├── worker.py
│   │   └── app.py                    <--- (2)
│   └── util
├── test
│   ├── main
│   │   ├── test_factory.py          <--- (3)
│   │   └── test_save.py             <--- (1)
└── misc
```

1. Geben Sie den absoluten Pfad zu `test_save.py` an:
2. Geben Sie den relativen Pfad von `src` zu `app.py` an:
3. Geben Sie den relativen Pfad von `factories` zu `test_factory.py` an:
4. Geben Sie das Kommando an, mit dem Sie den absoluten Pfad zu Ihrem aktuellen Arbeitsverzeichnis in der Shell ausgeben können.
5. Die Bash Shell ist ein...

<input type="checkbox"/> Booster	<input type="checkbox"/> Interpreter	<input type="checkbox"/> Alles davon
<input type="checkbox"/> Compiler	<input type="checkbox"/> Mixer	<input type="checkbox"/> Nichts davon