

# Klausurvorbereitung

---

## allgemeine Verständnisfragen

Es wird nicht verlangt, Romane zu schreiben. Sollte man sich für Stichpunkte entscheiden, dann müssen sie aussagekräftig statt schwammig sein. Bei Multiple-Choice-Fragen müssen genau die falschen Antworten durchgestrichen werden.

1. Nenne ein Beispiel für Hardware und ein Beispiel für Software.
2. Nenne Vor- und Nachteile zwischen Arbeitsspeicher und Festplatte. Erkläre, wieso Computer sowohl RAM als auch Festplattenspeicher besitzen.
3. Ein Computer hat folgende Hardwarekomponenten:
  - CPU mit vier Kernen (Quadcore) zu je 4,2 GHz
  - Zwei eingesetzte RAM-Riegel zu je 512 MB
  - HDD Festplatte mit 1 TB Speicherkapazität

Weshalb wird der Rechner trotz überdurchschnittlich schnellem Prozessor sehr sehr langsam laufen? Nenne mindestens zwei Optionen, um den Rechner zu verschnellern.

4. Nenne ein Programm, das live zeigt, wie sehr der RAM gefüllt ist und wie stark die CPU belastet ist.
5. Nenne wesentliche Unterschiede zwischen einer Verknüpfung und einer Kopie. Welche Konsequenzen ergeben sich daraus im Umgang mit der Originaldatei?
6. `high_res_image.bmp` ist ein qualitativ hochaufgelöstes und unkomprimiertes Bild. Es verbraucht 1 GB im Speicher.
  - i. Max hat eine 500 GB Festplatte. Welche der folgenden Szenarien sind technisch gesehen möglich?
    - Max will 499 Kopien erstellen
    - Max will 499 Verknüpfungen erstellen
  - ii. Max will das Bild löschen. Wann sind die Daten mit hoher Sicherheit, möglicherweise bzw. keinesfalls **physikalisch** von der Festplatte entfernt?
    - Gleich nachdem Max das Bild in den Papierkorb schmeißt.
    - Gleich nachdem Max den Papierkorb entleert hat.
    - Einen Monate nachdem Max den Papierkorb entleert hat.
    - Nachdem Max die Festplatte komplett überschreibt.
    - Nachdem Max die Festplatte herausnimmt und kräftig mit einem Hammer

draufschlägt.

7. Erkundige dich selbst: Was versteht man unter *IT-Forensik*? Was ist eine *forensische Kopie* einer Festplatte?
8. Was versteht man allgemein unter einer *Referenz*. Nenne außerdem zwei mögliche Beispiele für Referenzen in der Informatik.
9. Was versteht man in der Informatik unter dem Begriff des *Übersetzens*? Nenne die drei gängigen Arten von Übersetzern und grenze sie voneinander ab.
10. Handelt es sich bei NodeJS um einen Compiler oder um einen Interpreter? Wie heißt die Programmiersprache, die NodeJS übersetzt?
11. Softwarelizenzen
  - i. Wodurch ist Software unter OpenSource Lizenz bzw. proprietärer Lizenz im Wesentlichen charakterisiert?
  - ii. Nenne eine Möglichkeit, wie man mit OpenSource Software Geld verdienen kann.
  - iii. Wie bezeichnet man kostenlose proprietäre Software?
  - iv. Nenne jeweils ein Beispiel für OpenSource Software und proprietärer Software.
12. Nenne die beiden im Unterricht angesprochenen wichtigen Aufgaben eines Betriebssystems.
13. Was passiert beim Start eines Programms?
  - i. Alle für das Programm relevanten Informationen werden von der Festplatte in den Arbeitsspeicher geladen.
  - ii. Das Programm wird ausgeführt.
14. Was kann man tun, wenn ein Prozess hängt und der Rechner deswegen nur noch sehr langsam reagiert?
15. Was passiert, wenn man Software installiert?
  - i. Die für das Funktionieren des Programms wichtigen Dateien werden in die Festplatte geschrieben.
  - ii. Die für das Funktionieren des Programms wichtigen Dateien werden in den RAM geschrieben.
16. Wann wird ein Programm zu einem Prozess?

- i. Wenn man das Programm startet.
- ii. Wenn man die Software, die das Programm beinhaltet auf dem Rechner installiert.

17. Max bearbeitet gerade eine Textdatei. Dazu geht er folgendermaßen vor:

- i. Er öffnet die Datei.
- ii. Er schreibt die Zeichenkette "hallo" in die letzte Zeile.
- iii. Er speichert ab.
- iv. Er schließt die Textdatei.

An welchen Ereignissen von 1. - 4. wird von der Festplatte gelesen bzw. in die Festplatte geschrieben?

## "Rechenaufgaben"

Außer bei JavaScript-Code haben die Farben keine besondere Bedeutung, sondern wurden vom markdown-to-pdf-Konverter aus irgendeinem komischen Grund automatisch erzeugt ...

1. Bestimme die nächsten beiden Binärzahlen nach 00100110
2. Bestimme die Auswirkung des folgenden Assemblercodes auf den Speicher. Was muss außerdem mit dem Assemblercode geschehen, bevor er ausgeführt werden kann?

Speicher:

addr	584	585	586	587
hex	02	06	00	00

Assemblercode:

```
GETA $255, 584
LDB $0, $255, 0
LDB $1, $255, 1
ADD $0, $0, $1
STB $0, $255, 3
```

3. Angenommen, das komplette Dateisystem eines Windows-Rechners würde nur aus folgenden Dateien bestehen:

```
C:\system\winload.efi
C:\system\Ntoskrnl.exe
C:\system\Services.exe
```

```
C:\system\drivers\keyboard.dll
C:\system\drivers\monitor.dll
C:\util\cmd.exe
C:\util\textedit.exe
C:\documents\hallo.txt
```

- i. Zeichne dieses Dateisystem als Baum.
- ii. Benenne die absoluten Pfade aller Ordner.

#### 4. Max programmiert folgenden JavaScript-Code

```
var text = "hallo";
console.log(text);
```

und speichert diesen Code in einer Datei namens `hallo.js` im Verzeichnis

`C:\Users\Max\mein_erstes_Softwareprojekt\` ab.

Max startet eine Konsole mit `cmd.exe` und will node mit dem Skript aufrufen. Also tippt er folgendes ein:

```
C:\Users\Max\> node hallo.js
```

Jedoch erscheint die Fehlermeldung

```
module.js:471
  throw err;
  ^

Error: Cannot find module 'C:\Users\Max\hallo.js'
    at Function.Module._resolveFilename (module.js:469:15)
    at Function.Module._load (module.js:417:25)
    at Module.runMain (module.js:604:10)
    at run (bootstrap_node.js:383:7)
    at startup (bootstrap_node.js:149:9)
    at bootstrap_node.js:496:3
```

Korrigiere Max's Fehler.

5. Max will die Werte zweier Variablen tauschen. Erkläre den Fehler in seinem Code und korrigiere.

```
a = b;  
b = a;
```

6. Max programmiert aus Lust und Laune einfach mal folgendes Skript.

```
var text = '';  
while(true) {  
  text = text + 'a';  
}
```

Wie viel Speicherplatz verursacht das Skript bei Nichtausführung bzw. Ausführung?

- i. weniger als 1 Megabyte auf der Festplatte
  - ii. mehr als 1 Megabyte auf der Festplatte
  - iii. mehr als 1 Gigabyte auf der Festplatte
  - iv. weniger als 1 Megabyte auf dem Arbeitsspeicher
  - v. mehr als 1 Megabyte auf dem Arbeitsspeicher
  - vi. mehr als 1 Gigabyte auf dem Arbeitsspeicher
7. Notiere die grundsätzliche Syntax einer vorprüfenden Schleife in JavaScript.
8. Notiere die grundsätzliche Syntax einer Verzweigung mit Alternative in JavaScript.
9. Schreibe eine Funktion, die zwei Zahlen als Parameter erhält und die größere zurückgibt.

```
function max(x,y) {  
  // ... dein code ...  
}
```

10. Schreibe eine Funktion, die testet, ob ein String ein Leerzeichen enthält oder nicht. Überlege dir selbst, was diese Funktion zurückgeben muss.

```
function hat_space(zeichenkette) {  
  // ... dein code ...  
}
```

11. Schreibe eine Funktion, die zählt, wie viele Leerzeichen ein String enthält. Überlege dir selbst, was diese Funktion zurückgeben muss.

```
function anzahl_spaces(zeichenkette) {  
  // ... dein code ...  
}
```