

Abi Checkliste Codierung

- [Bildungsplan Codierung](#)

Leitfragen

- Wiki: <https://info-bw.de/faecher:informatik:oberstufe:codierung:start>

Zahlen

- Binärsystem, Hexadezimalsystem, Hexadezimale zahlen als als Abkürzung für 4 binäre Bits, Umrechnungen zwischen den Zahlensystemen.
- Oktalsystem: Eine Stelle eine Oktalzahl als Abkürzung für 3 binäre Bits.
- Ganze Zahlen: Lösungsansätze um negative Zahlen darstellen zu können, Zweierkomplementdarstellung als beste Lösung (Warum?).
- Rechnen mit Zahlen im Zweierkomplement, vor allem Addition und Subtraktion durch Addition der Gegenzahl (Bestimmung der Gegenzahl!). Einfache Multiplikationen sind denkbar, aber eher unwahrscheinlich.
- Festkommazahlen: Wie funktioniert, was sind die Probleme?
- Fließkommazahlen sind nicht Inhalt des Bildungsplans, kommen also nicht dran

Zeichen

- Was ist ASCII - warum reicht das nicht?
- Wie wurde das früher erweitert? Was sind Codepages?
- Welches Ziel verfolgt die UTF8 Codierung von Zeichen?
- Was ist der Unterschied zwischen einem Codepunkt ("Zeichenummer") und der Codierung. Das Modell des UFT-8 Zugs hilft hier unter Umständen weiter.
- Welche Probleme treten auf, wenn die Codierung und die Decodierung von Zeichen nicht zueinander passen? Wie äußert sich das in der realen Welt?

Grafik

- Was ist ein Pixel? Was eine Pixelgrafik? Nachteile? Vorteile?
- Was ist eine Vektorgrafik (am Beispiel SVG kann man das gut erklären). SVG Syntax muss man aber nicht auswendig können, es geht nur um die unterschiedlichen Prinzipien von Pixel- und Vektorgrafiken und die daraus folgenden Konsequenzen.
- Was ist ein "Datagramm".
- Wie kann man Farben darstellen? - Welche Verbindung zu hexadezimalen Zahlen besteht hier.
- Du solltest ein Datagramm decodieren oder encodieren können, wenn die Regeln angegeben sind, z.B. so etwas:
<https://info-bw.de/faecher:informatik:oberstufe:codierung:bilder:paletten:start>

Kompression

- Benötigt man Kompression heutzutage überhaupt noch?
- Was ist der Unterschied zwischen verlustfreier und verlustbehafteter Kompression?
- Lauflängencodierung: wie kann man das machen (verschiedene Möglichkeiten), wann funktioniert es gut, wann schlecht?
- **Huffman**-Codierung:
 - Wie funktioniert der Algorithmus. Wie sieht ein gültiger Huffman-Baum aus?
 - Wie kann man aus dem Baum die Codierung ableiten?
 - Wie kann man ein Datagramm decodieren, wenn man die Codierung/den Baum hat?
Aufgaben auf der Wikiseite.
- Erläutere das Vorgehen beim **LZW** Verfahren - welche beiden wesentlichen Vorteile gegenüber Huffman hat das Verfahren?
- Du solltest einfache Informationen von Hand mit LZW Codieren können.
- Aufgaben im Wiki, wichtig die Frage 6: "Begründe, dass das LZW-Verfahren nicht jede Eingabe komprimieren kann."

From:
<https://www.info-bw.de/> -

Permanent link:
<https://www.info-bw.de/faecher:informatik:oberstufe:glossar:abocodierung:start>

Last update: **22.04.2025 08:07**

