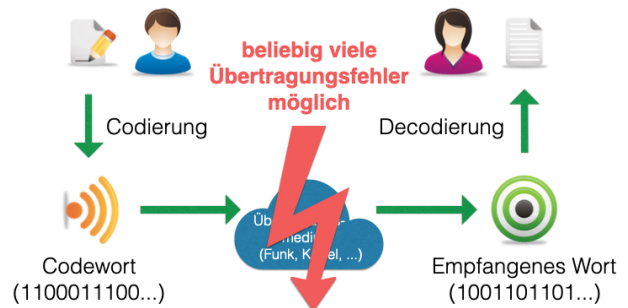


## Informationen, Daten und Codierung Fehlerkorrigierende Codes II

Die folgende Tabelle zeigt noch einmal die Codierung der vier möglichen Nachrichten aus zwei Bits:

Nachricht	Codewort
00	000000
01	010111
10	101011
11	111100



Schaut man sich zwei Codewörter an (z.B. 010111 und 101011), so stimmen sie in einigen Bits überein und in anderen unterscheiden sie sich. Im Beispiel oben sind die hinteren zwei Bits gleich, der Rest ist verschieden. Die **Anzahl** der Bits, in denen sich zwei Codewörter A und B unterscheiden, nennt man „*Hamming-Distanz zwischen A und B*“. Die beiden Codewörter im Beispiel sind in 4 Bits verschieden, also ist ihrer Hamming-Distanz gleich 4 (man schreibt  $d(010111, 101011) = 4$ ).

Es könnte nun durch genau 4 Übertragungsfehler passieren, dass aus dem Codewort 010111 das Codewort 101011 wird, ohne dass dies auffällt (das empfangene Wort ist ja wieder ein gültiges Wort). Treten *weniger* als 4 Fehler auf, so entstehen Worte, die nicht gültig sind. Der Empfänger erkennt also, dass die Übertragung fehlerhaft war (Abb. 2 zeigt die beiden gültigen Codewörter (schwarz) und drei ungültige (weiß) dazwischen).

Tritt nur *ein* Übertragungsfehler auf (z.B. wird 010111 zu 110111), so wird der Fehler erkannt und kann sogar korrigiert werden. Bei zwei Fehlern (z.B. wird 010111 zu 100111) ist der Fehler nicht korrigierbar! Ist dir klar, warum?

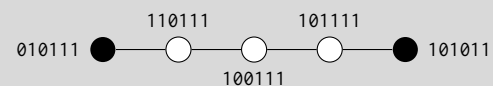


Abb. 2: Ungültige Worte zwischen 010111 und 101011

### Aufgabe 1

Berechne auch für die übrigen Kombinationen von Codewörtern aus der Tabelle oben die Hamming-Distanz. Was fällt dir auf?

Was bedeutet es praktisch, wenn zwei Codewörter die Hamming-Distanz 1 haben?

### Aufgabe 2

Als Empfänger erhältst du das Wort 111011. Dieses Wort ist offensichtlich falsch übertragen worden. Als welches Codewort würdest du es interpretieren und wie lautet somit die gesendete Nachricht?

### Aufgabe 3

Als Empfänger erhältst du das Wort 111111. Dieses Wort gehört nicht zu den gültigen Codewörtern, es ist also ein Übertragungsfehler aufgetreten. Begründe, warum das Wort nicht sicher korrigiert werden kann.

### ★ Zusatzaufgabe 1

Es ist nicht besonders effizient, für **zwei** Nachrichtenbits **vier** Prüfbits zu versenden. Kannst du einen Code für die vier 2-Bit Nachrichten von oben entwerfen, der mit *weniger* als vier Prüfbits auskommt, aber trotzdem einen Fehler korrigieren kann? Wie viele Übertragungsfehler kann dein Code erkennen?