

Selbstkorrigierender Hamming-Code

An einigen Stellen, wie z. B. auf CDs ist es wichtig, nicht nur die Fehler in Daten zu erkennen, sondern sie sogar korrigieren zu können. Dieses soll an binären Zahlen überprüft werden. In einer ersten Überlegung könnte man die Daten einfach doppelt aufschreiben. Aus 010 wird dann 001100.

1. Gib am Beispiel 000010 begründet an, dass diese Variante nicht sinnvoll ist.

Statt doppelt kann man die Zeichen auch verdreifachen. So wird aus dem Beispiel 010 der Code 000111000. Damit lassen sich Fehler an einer einzelner Ziffer erkennen und auch beheben.

2. Gib an, welcher Wert hier übermittelt werden sollte:

- a) 010000110111
- b) 111011000101

Bei dieser Methode muss aber die dreifache Menge an Daten gespeichert werden um einfache Fehler zu finden. Richard Hamming (1915 – 1998) hatte dafür eine passende Idee, aus der er den Hamming-Code entwickelte. Er schreibt für jeweils vier Bit insgesamt sieben Bit nach folgendem Muster:

Nachrichten		Codewörter
1000	→	1110000
0100	→	1001100
0010	→	0101010
0001	→	1101001

Alle anderen Möglichkeiten werden dadurch gebildet, dass die Codewörter entsprechend miteinander kombiniert werden. So wird aus einer 0 und 0 oder einer 1 und 1 an der gleichen eine 0. Aus 1 und 0 wird eine 1. So ist für 1001 der Code 0011001.

3. Gib für alle anderen Kombinationen die genauen Codewörter an.

Wird ein Bit falsch übermittelt, so kann man dadurch auf die richtigen Daten schließen, indem man den empfangenen Code mit der Liste vergleicht. Das Codewort, das die wenigsten Abweichungen hat ist das zugehörige. So ist bei 0100111 das mit den wenigsten Änderungen die 0100101.

4. Suche heraus, welche Daten durch folgende Codewörter übermittelt werden:

- a) 1011011
- b) 0110101
- c) 0100010

5. Tausche mit deinen Mitschülern Codewörter aus, deren Werte sie ermitteln müssen.



Lösungen:

Selbstkorrigierender Hamming-Code:

1.

2.

3. Alle Codewörter

1111 → 1111111
1110 → 0010110
1101 → 1010101
1100 → 0111100
1011 → 0110011
1010 → 1011010
1001 → 0011001
1000 → 1110000
0111 → 0001111
0110 → 1100110
0101 → 0100101
0100 → 1001100
0011 → 1000011
0010 → 0101010
0001 → 1101001

4. a) 1010

b) 0101

c) 0010

