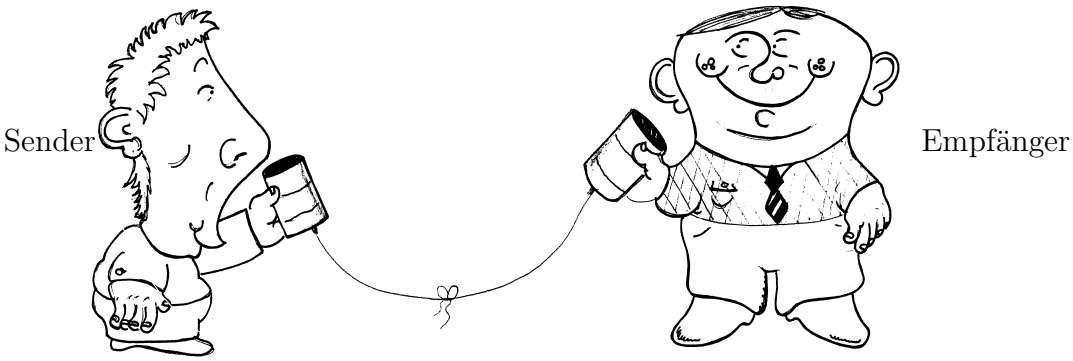
1 Kryptographie und Codierungstheorie

Zunächst sind die Begriffe „Kryptographie“ und „Codierungstheorie“ zu unterscheiden, da sie häufig vermengt, nicht scharf voneinander getrennt oder gar synonym gebraucht werden.
Die Kryptographie versucht die Übertragung einer Nachricht vor Dritten geheim zu halten und die Nachricht auch vor Veränderung durch Dritte zu sichern.
Die Codierungstheorie versucht Übertragungsfehler zu erkennen und evtl. sogar auszubessern.
Um die jeweiligen Ziele zu erreichen, muss aus dem Klartext ein neuer Text erzeugt werden — ein Geheimtext oder ein codierter Text.



Knoten, der die
Übertragung stört

z.B. Zufall, Unvorsichtigkeit,
böser Mensch, Sabotage

Man spricht von Texten, möchte aber mathematische Verfahren anwenden, daher muss eine Unwandlung des Textes in Zahlen vorgenommen werden.
Um mit Buchstaben mathematisch umzugehen werden sie mit Zahlen identifiziert.

Beispiel 1.

i) Man betrachtet nur die großen Buchstaben und ordnet ihnen die Zahlen von 01 *bis* 26 *zu.*

ii) Eine übliche Norm stellt die ASCII-Codierung dar (vgl. Blatt).

2 Einführungsbeispiel: ISB-Nummer

„Leider“ ist die 10-stellige ISBN (Internationale Standard-BuchNummer) nicht mehr in vollem Umfang in Gebrauch, da seit Januar 2007 auf eine 13-stellige ISBN umgestellt wird. Die 13-stellige ISBN unterscheidet sich wesentlich von der 10-stelligen ISBN, aber nicht von den üblichen Artikelnummern EAN und ist überdies weniger leistungsfähig. Deshalb wird an dieser Stelle die 10-stellige ISBN bevorzugt, um insbesondere auch Betrachtungen zur Leistungsfähigkeit anstellen zu können.
Das Buch „Die Physiker“ von Friedrich Dürrenmatt, im Diogenes Verlag erschienen, hat folgende Internationale Standard-BuchNummer (ISBN):

$$\underbrace{3}_{a} - \underbrace{257}_{b} - \underbrace{23047}_{c} - \underbrace{8}_{d}$$

a) Gruppennummer, d.h. Land (hier: deutscher Sprachraum)

b) Verlagsnummer (hier: Diogenes Verlag)

c) Titelnummer (hier: „Die Physiker“ von Friedrich Dürrenmatt)

d) Prüfziffer

Die ISBN ist stets 10-stellig; die Prüfziffer immer einstellig. Bei Verlags- und Titelnummer können dagegen die Anzahlen der Ziffern variieren.

Aufgabe 2. *Warum haben Verlags- und Titelnummer keine feste Ziffernanzahl?*

Die Codierung

Dürrenmatt „Die Physiker“	$\longmapsto$	23047
Diogenes	$\longmapsto$	257
deutsch	$\longmapsto$	3

enthält bereits alle relevanten Informationen

Die Prüfziffer (hier: 8) beinhaltet keine neue Information. Es wird also mehr übermittelt als eigentlich nötig wäre.

Beispiel aus dem Alltag:

Übermittlung einer Kontonummer über das Telefon.

Die Nummer wird **mehrmals** von Sender oder Empfänger vorgelesen.

Berechnung der Prüfziffer (Methode 1)

Die Ziffern der ISBN werden mit den Ziffern 1 bis 10 in absteigender Reihenfolge multipliziert:

$10 \cdot 3 + 9 \cdot 2 + 8 \cdot 5 + 7 \cdot 7 + 6 \cdot 2 + 5 \cdot 3 + 4 \cdot 0 + 3 \cdot 4 + 2 \cdot 7 =$

$= 30 + 18 + 40 + 49 + 12 + 15 + 0 + 12 + 14 = 190$

Man sucht die nächstgrößere durch 11 teilbare Zahl: sie ist hier 198.

Die Prüfziffer ergibt sich als Differenz:

$$p = 198 - 190 = 8$$

Eine 10-stellige Ziffernfolge mit einer auf diese Weise berechneten Prüfziffer nennt man gültige ISBN (im Unterschied zu ungültigen ISBN).

Hinweis 3. *Bekannte Teilbarkeitsregeln*

i) Mit Endziffernregeln kann man Zahlen auf die Teilbarkeit durch 2, 4 und 8 prüfen.

ii) Ebenso gibt es Endziffernregeln für die Teilbarkeit durch 5, 25 und 125.

iii) Die Quersummenregel gibt Aufschluss über die Teilbarkeit durch 3 und 9.

iv) Mit der alternierenden Quersummenregel kann die Teilbarkeit durch 11 geprüft werden.

Bemerkung 4. *Eine Zahl* $a_n a_{n-1} \cdots a_2 a_1 a_0 = \sum_{\nu=0}^{n} a_\nu \cdot 10^\nu$ *ist genau dann durch 11 teilbar, wenn die altenierende Quersumme* $a_0 - a_1 + a_2 - a_3 + (-1)^4 a_4 + (-1)^5 a_5 + \cdots + (-1)^n a_n = \sum_{\nu=0}^{n} (-1)^\nu a_\nu$ *durch 11 teilbar ist.*

Beispiel 5.

i) 328591 ist nicht durch 11 teilbar, da $-3+2-8+5-9+1=-12$ *nicht durch 11 teilbar ist.*

ii) 1234554321 ist durch 11 teilbar, da die altenierende Quersumme 0 ist.

Beispiel 6.

i) $\sum_{\nu=0}^{n} 3^\nu = 1+3+9+27+3^4+\cdots+3^n$

ii) $\sum_{\nu=0}^{n} \nu^2 = 0+1+4+9+4^2+\cdots+n^2$

iii) $\sum_{\nu=2}^{7}(\nu+1) = (2+1)+(3+1)+(4+1)+(5+1)+(6+1)+(7+1) = 33$

iv) $\sum_{\nu=0}^{n} 1 = 1+1+\cdots+1 = n+1$

Aufgaben 7.

i) Ist 4-73845-123-4 eine gültige ISBN?

ii) Bestimme die gültige Prüfziffer p in der ISBN 8-828-32123-p.

iii) Finde ein x so, dass x-323-42185-3 eine gültige ISBN ist. Aus welchem „Land“ kommt das Buch?

iv) Finde ein x so, dass x-323-42185-7 eine gültige ISBN ist. Welches Problem ist aufgetreten?

Beispiel 8. *Das Buch „Topologikon“ von Jean-Pierre Petit erschienen bei Vieweg hat die ISBN*

$$3-528-06675-\mathrm{X}.$$

Als Prüfziffer ergibt sich nach obiger Methode 10. Man behilft sich nun mit der 10 der Römer, dem X. Nur an dieser Stelle ist eine 11-te Ziffer möglich.

Berechnung der Prüfziffer (Methode 2):

Bei der ISBN des „Topologikons“ $3-528-06675-\mathrm{X}$ könnte man auch bei der Multiplikation mit 1 beginnen

$1 \cdot 3 + 2 \cdot 5 + 3 \cdot 2 + 4 \cdot 8 + 5 \cdot 0 + 6 \cdot 6 + 7 \cdot 6 + 8 \cdot 7 + 9 \cdot 5 =$

$= 3 + 10 + 6 + 32 + 0 + 36 + 42 + 56 + 45 = 230$

Nun dividiert man 230 durch 11 und erhält:

$$230 : 11 = 22 \quad \text{Rest} \quad 10.$$

Die Prüfziffer 10 = X erhält man nun als Rest bei der Division durch 11.

Eine Erleichterung bietet das sog. Modulorechnen.

Wir machen uns dazu lediglich ganz anschaulich mit den Rechenregeln bekannt:

Definition 9. *Sei $n \in \mathbb{N}$ und $x, y \in \mathbb{Z}$. Man sagt, dass x und y kongruent modulo n sind, wenn x und y bei der Division durch n den selben Rest lassen; man schreibt kurz:*

$$x \equiv y \mod n.$$

Beispiel 10.

i) $17 \equiv 12 \mod 5$, *da* $17 : 5 = 3$ *Rest 2 und* $12 : 5 = 2$ *Rest 2*

ii) $37 \equiv 1 \mod 36$, *da* $37 : 36 = 1$ *Rest 1*

iii) $17 \equiv 17 + 7 \mod 7 \equiv 24 \mod 7$, *da* $17 : 7 = 2$ *Rest 3 und* $(17 + 7) : 7 = 17 : 7 + 7 : 7 = 3$ *Rest 3*

iv) $25 \equiv 0 \mod 5$, *d.h. 5 teilt 25 (kurz:* $5|25$*)*

v) $-2 \equiv 3 \mod 5$, *denn* $-2 : 5 = (-1) \cdot 5 \underline{+} 3$

Bemerkung 11. *Äquivalent sind:*

i) $x \equiv y \mod n$

ii) $n|(x - y)$

Ist insbesondere $x \equiv 0 \mod n$, so folgt $n|x$.

Satz 12. *Seien $x, y, a, b \in \mathbb{Z}$ und $m, n \in \mathbb{N}$ mit $x \equiv y \mod n$ und $a \equiv b \mod n$. Es gilt:*

i) $x \pm a \equiv y \pm a \mod n$

ii) $x \pm a \equiv y \pm b \mod n$

iii) $x \cdot a \equiv y \cdot a \mod n$

iv) $x \cdot a \equiv y \cdot b \mod n$

v) $x^m \equiv y^m \mod n$

Beispiel 13. *Aus* $3 \equiv 5 \mod 2$ *und* $1 \equiv 7 \mod 2$ *folgt:*

i) $3 \pm 6 \equiv 5 \pm 6 \mod 2$

ii) $3 \pm 1 \equiv 5 \pm 7 \mod 2$

iii) $3 \cdot 6 \equiv 5 \cdot 6 \mod 2$

iv) $3 \cdot 1 \equiv 3 \cdot 7 \mod 2$

v) $3^4 \equiv 5^4 \mod 2$

Aber: *aus* $3 \cdot 6 \equiv 4 \cdot 6 \mod 2$ *folgt nicht* $3 \equiv 4 \mod 2$

Bemerkung 14. *Ist* $\mathrm{ggT}(m,n) = 1$*, so folgt aus* $x \cdot m \equiv y \cdot m \mod n$ *auch* $x \equiv y \mod n$*. Ist* $\mathrm{ggT}(m,n) \neq 1$*, so folgt aus* $x \cdot m \equiv y \cdot m \mod n$ *lediglich* $x \equiv y \mod \mathrm{ggT}(m,n)$*.*

Bemerkung 15. *Die Menge aller Reste bildet bzgl. der Addition eine abelsche Gruppe (bzgl. Addition und Multiplikation einen Ring), falls n eine Primzahl ist, ergibt sich sogar ein Körper.*
Wir schreiben:

$$\mathbb{Z}/n\mathbb{Z} = \{\overline{0}, \overline{1}, \cdots \overline{n-1}\}$$

und rechnen etwa in $\mathbb{Z}/3\mathbb{Z}$*:*
$\overline{2} + \overline{1} = \overline{0}$
$\overline{2} \cdot \overline{2} = \overline{1}$
Nochmals: *Vorsicht bei der Division!*
Häufig lassen wir auch den Querstrich über den Zahlen weg.

Aufgabe 16. *Gib jeweils alle natürlichen Zahlen ($\mathbb{N}_0$) an, die die nachstehenden Kongruenzen lösen.*

i) $x \equiv 5 \mod 11$

ii) $x + 2 \equiv 3 \mod 17$

iii) $x - 7 \equiv 5 \mod 13$

iv) $2 \cdot x \equiv 4 \mod 17$

v) $2 \cdot x \equiv 4 \mod 16$

vi) $x : 2 \equiv 4 \mod 17$

vii) $x : 2 \equiv 1 \mod 9$

viii) $x \equiv 2 \mod 9$

ix) $x \equiv 1 \mod 3 \quad \wedge \quad x \equiv 2 \mod 5$

x) $x \equiv 2 \mod 4 \quad \wedge \quad x \equiv 3 \mod 9$

xi) $x \equiv 1437289174 \mod 11$

xii) $x \equiv 123456789 \mod 11$

xiii) $x \equiv 123459999 \mod 9$

xiv) $x \equiv 13^2 \mod 10$

xv) $2 \cdot x \equiv 34567899344 \mod 3$

xvi) $11 \cdot x \equiv 111111111 \mod 11$

xvii) $11 \cdot x \equiv 11111111 \mod 11$

xviii) $x^2 \equiv 3 \mod 10$

xix) Welche Ziffern sind an der Einerstelle bei Quadratzahlen möglich, welche nicht?

Nun wieder zur ISBN: $a_1a_2a_3a_4a_5a_6a_7a_8a_9 - a_{10}$.
Man berechnet nach Methode 1

$$10 \cdot a_1 + 9 \cdot a_2 + 8 \cdot a_3 + 7 \cdot a_4 + 6 \cdot a_5 + 5 \cdot a_6 + 4 \cdot a_7 + 3 \cdot a_8 + 2 \cdot a_9$$

und sucht $a_{10} \in \{0, 1, 2, \cdots, 10\}$, sodass

$$10 \cdot a_1 + 9 \cdot a_2 + 8 \cdot a_3 + 7 \cdot a_4 + 6 \cdot a_5 + 5 \cdot a_6 + 4 \cdot a_7 + 3 \cdot a_8 + 2 \cdot a_9 + 1 \cdot a_{10} \equiv 0 \mod 11.$$

Diese Gleichung nennt man Prüfgleichung für die ISBN.
Wir stellen fest, dass die Summe aus Vorfaktor und Index stets 11 ergibt, z.B. bei $3 \cdot a_8$ ist $3 + 8 = 11$.
Ein kleiner Trick hilft die Prüfgleichung einfacher aufzuschreiben und das ist auch der erste Schritt zu zeigen, dass beide vorgestellten Methoden äquivalent sind.
Multipliziert man eine Zahl mit 11, ist diese stets durch 11 teilbar:

$$\begin{aligned} &11a_1 + 11a_2 + 11a_3 + 11a_4 + 11a_5 + 11a_6 + 11a_7 + 11a_8 + 11a_9 + 11a_{10} \\ &\equiv 0 \mod 11. \end{aligned}$$

Subtrahiert man davon die obige Prüfgleichung ergibt sich:

$$\begin{aligned} &(11 - 10) \cdot a_1 + (11 - 9) \cdot a_2 + (11 - 8) \cdot a_3 + (11 - 7) \cdot a_4 + (11 - 6) \cdot a_5 \\ &+(11 - 5) \cdot a_6 + (11 - 4) \cdot a_7 + (11 - 3) \cdot a_8 + (11 - 2) \cdot a_9 + (11 - 1) \cdot a_{10} \\ &\equiv 0 \mod 11 \end{aligned}$$

und somit eine andere Schreibweise für die Prüfgleichung der ISBN:

$$1 \cdot a_1 + 2 \cdot a_2 + 3 \cdot a_3 + 4 \cdot a_4 + 5 \cdot a_5 + 6 \cdot a_6 + 7 \cdot a_7 + 8 \cdot a_8 + 9 \cdot a_9 + 10 \cdot a_{10} \equiv 0 \mod 11.$$

$$\begin{aligned} &1 \cdot a_1 + 2 \cdot a_2 + 3 \cdot a_3 + 4 \cdot a_4 + 5 \cdot a_5 + 6 \cdot a_6 + 7 \cdot a_7 + 8 \cdot a_8 + 9 \cdot a_9 + 10 \cdot a_{10} \\ &= \sum_{\nu=1}^{10} \nu \cdot a_\nu \equiv 0 \mod 11. \end{aligned}$$

Aufgaben 17.

i) Schreibe die andere Prüfgleichung mit Hilfe des Summenzeichens.

ii) Schreibe die obige Herleitung der alternativen Prüfgleichung mit Hilfe des Summenzeichens.

iii) Man überlege sich Rechenregeln für das Summenzeichen.

Mit der neuen Prüfgleichung kann man die Prüfziffer ganz leicht bestimmen:

$$\begin{aligned} \sum_{\nu=1}^{10} \nu \cdot a_\nu &\equiv 0 \mod 11 \\ a_{10} + \sum_{\nu=1}^{10} \nu \cdot a_\nu &\equiv a_{10} + 0 \mod 11 \\ (1+10) \cdot a_{10} + \sum_{\nu=1}^{9} \nu \cdot a_\nu &\equiv a_{10} \mod 11 \\ a_{10} &\equiv \sum_{\nu=1}^{9} \nu \cdot a_\nu \mod 11. \end{aligned}$$

Die Prüfziffer ist also der Rest bei der Divison durch 11 von

$$\sum_{\nu=1}^{9} \nu \cdot a_\nu = 1 \cdot a_1 + 2 \cdot a_2 + 3 \cdot a_3 + 4 \cdot a_4 + 5 \cdot a_5 + 6 \cdot a_6 + 7 \cdot a_7 + 8 \cdot a_8 + 9 \cdot a_9.$$

Fehler, die etwa bei der Buchbestellung auftreten, wenn die ISBN auf einer Tastatur eingegeben wird:
Verwechslung einer Ziffer (Einzelfehler) stellen 79 % aller Fehler dar.
Vertauschung benachbarter Ziffern (Ziffern-Dreher) sind 10 % aller Fehler.
Man erkennt mit Hilfe der ISBN, dass ein Fehler vorliegt, auf zwei Arten:

i) die Berechnung der Prüfziffer mit einer der beiden Methoden liefert ein anderes Ergebnis.

ii) die Prüfgleichung (nach Methode 1 oder 2) ist nicht erfüllt.

Beispiel 18. *Ist folgende ISBN gültig?*

$$0 - 19 - 853804 - 9$$

i) $1 \cdot 0 + 2 \cdot 1 + 3 \cdot 9 + 4 \cdot 8 + 5 \cdot 5 + 6 \cdot 3 + 7 \cdot 8 + 8 \cdot 0 + 9 \cdot 4 = 196 \equiv$
$\equiv 9 \mod 11.$
Die ISBN ist gültig, da sich die richtige Prüfziffer 9 ergibt.

ii) $1 \cdot 0 + 2 \cdot 1 + 3 \cdot 9 + 4 \cdot 8 + 5 \cdot 5 + 6 \cdot 3 + 7 \cdot 8 + 8 \cdot 0 + 9 \cdot 4 + 10 \cdot 9 = 286 \equiv$
$\equiv 0 \mod 11$ oder
$10 \cdot 0 + 9 \cdot 1 + 8 \cdot 9 + 7 \cdot 8 + 6 \cdot 5 + 5 \cdot 3 + 4 \cdot 8 + 3 \cdot 0 + 2 \cdot 4 + 1 \cdot 9 = 231 \equiv$
$\equiv 0 \mod 11.$
Die ISBN ist gültig, da die Prüfgleichung erfüllt ist.

Satz 19. *Das ISBN-Verfahren erkennt Einzelfehler.*

Beweis: Sei $a_1a_2a_3a_4a_5a_6a_7a_8a_9 - a_{10}$ die originale und gültige ISBN und $a'_1a'_2a'_3a'_4a'_5a'_6a'_7a'_8a'_9 - a'_{10}$ die an der i-ten Stelle verwechselte ISBN.
Zu zeigen ist, dass $a'_1a'_2a'_3a'_4a'_5a'_6a'_7a'_8a'_9 - a'_{10}$ ungültig ist.
Beweis durch Widerspruch:
Angenommen, trotz der Verwechslung einer Ziffer ist auch die zweite Prüfgleichung erfüllt, dann gilt:

$$1 \cdot a_1 + 2 \cdot a_2 + 3 \cdot a_3 + 4 \cdot a_4 + 5 \cdot a_5 + 6 \cdot a_6 + 7 \cdot a_7 + 8 \cdot a_8 + 9 \cdot a_9 + 10 \cdot a_{10} \equiv 0 \mod 11$$

und

$$1 \cdot a'_1 + 2 \cdot a'_2 + 3 \cdot a'_3 + 4 \cdot a'_4 + 5 \cdot a'_5 + 6 \cdot a'_6 + 7 \cdot a'_7 + 8 \cdot a'_8 + 9 \cdot a'_9 + 10 \cdot a'_{10} \equiv 0 \mod 11.$$

Da lediglich an der i-ten Stelle ein Fehler vorliegt, ergibt die Subtraktion:

$$i \cdot a_i - i \cdot a'_i \equiv 0 \mod 11$$

und somit

$$i \cdot (a_i - a'_i) \equiv 0 \mod 11.$$

Da $\text{ggT}(i, 11) = 1$, kann dies nach Bemerkung 14 nur erfüllt sein, falls $a_i - a'_i \equiv 0 \mod 11$. Da $a_i, a'_i \in \{0, 1, \cdots 10\}$, kann dies nur gelten, falls $a_i = a'_i$. Das kann aber nicht sein, was zu einem Widerspruch führt.
Somit ist die zweite (verwechselte) ISBN nicht gültig. □

Beispiel 20. $3-257-23047-8$ *ist eine gültige ISBN und*
$3-259-23047-8$ *ist wohl fehlerhaft.*
Angenommen, beide Prüfgleichungen wären dennoch erfüllt, dann wäre
$1\cdot 3+2\cdot 2+3\cdot 5+4\cdot 7+5\cdot 2+6\cdot 3+7\cdot 0+8\cdot 4+9\cdot 7+10\cdot 8\equiv 0 \mod 11$
und
$1\cdot 3+2\cdot 2+3\cdot 5+4\cdot 9+5\cdot 2+6\cdot 3+7\cdot 0+8\cdot 4+9\cdot 7+10\cdot 8\equiv 0 \mod 11$
und somit gilt für die Differenz

$$\begin{aligned} 1\cdot 3-1\cdot 3+2\cdot 2-2\cdot 2+3\cdot 5-3\cdot 5+4\cdot 9-4\cdot 7\cdots &\equiv 0 \mod 11\\ 4\cdot(9-7) &\equiv 0 \mod 11. \end{aligned}$$

Da $\mathrm{ggT}(4,11)=1$*, ergibt sich* $2\equiv 0 \mod 11$ *— ein Widerspruch.*

Neben den Einzelfehlern erkennt das Prüfziffernverfahren der ISBN auch noch beliebige Drehfehler (nicht nur Nachbarschaftsvertauschungen) — ganz im Unterschied zu den Verfahren der 13-stelligen ISBN, der EAN oder Kontonummern.
Der Nachweis erfolgt wiederum auf ganz ähnliche Weise mit einem Widerspruchsbeweis.

Satz 21. *Das ISBN-Verfahren erkennt beliebige Drehfehler (nicht nur die Vertauschung benachbarter Ziffern).*

Beweis: Sei $a_1a_2a_3a_4a_5a_6a_7a_8a_9-a_{10}$ die originale und gültige ISBN und $a'_1a'_2a'_3a'_4a'_5a'_6a'_7a'_8a'_9-a'_{10}$ die an der i-te und der j-te Stelle vertauschte ISBN ($i\neq j$). Auch ist $a_i\neq a_j$.
Zu zeigen ist, dass $a'_1a'_2a'_3a'_4a'_5a'_6a'_7a'_8a'_9-a'_{10}$ ungültig ist.
Beweis wieder durch Widerspruch:
Angenommen, trotz der Vertauschung sind beide Prüfgleichungen erfüllt, dann gilt:

$$\sum_{\nu=1}^{10}\nu\cdot a_\nu\equiv 0 \mod 11 \qquad \text{und} \qquad \sum_{\nu=1}^{10}\nu\cdot a'_\nu\equiv 0 \mod 11.$$

Da lediglich die i-te und die j-te Stelle vertauscht sind, ergibt die Subtraktion:

$$i\cdot a_i-i\cdot a'_i+j\cdot a_j-j\cdot a'_j\equiv 0 \mod 11.$$

Da aufgrund der Vertauschung $a_i = a'_j$ und $a_j = a'_i$, folgt

$$\begin{aligned} i \cdot (a_i - a_j) + j \cdot (a_j - a_i) &\equiv 0 \mod 11 \\ i \cdot (a_i - a_j) - j \cdot (a_i - a_j) &\equiv 0 \mod 11 \\ (i - j) \cdot (a_i - a_j) &\equiv 0 \mod 11. \end{aligned}$$

Da $i \neq j \in \{1, 2, 3, \cdots 10\}$, ist $-9 \leq i-j \leq 9$ und $i-j \neq 0$. Somit ist $\text{ggT}(i-j, 11) = 1$ und deshalb $a_i - a_j \equiv 0 \mod 11$ nach Bemerkung 14. Da $a_i, a_j \in \{0, 1, \cdots 10\}$, kann dies nur gelten, falls $a_i = a_j$. Damit liegt aber gar keine Vertauschung vor, was zu einem Widerspruch führt.
Somit ist die zweite ISBN nicht gültig. □

Ebenfalls kann an einem konkreten Zahlenbeispiel die Idee des obigen Widerspruchsbeweises klar gemacht werden.

Beispiel 22. $5 - 528 - 06675 - X$ *ist eine gültige ISBN und*
$5 - 628 - 05675 - X$ *ist fehlerhaft.*
Angenommen, beide Prüfgleichungen wären erfüllt, dann wäre
$1 \cdot 5 + 2 \cdot 5 + 3 \cdot 2 + 4 \cdot 8 + 5 \cdot 0 + 6 \cdot 6 + 7 \cdot 6 + 8 \cdot 7 + 9 \cdot 5 + 10 \cdot 10 \equiv$
$\equiv 0 \mod 11$
und
$1 \cdot 5 + 2 \cdot 6 + 3 \cdot 2 + 4 \cdot 8 + 5 \cdot 0 + 6 \cdot 5 + 7 \cdot 6 + 8 \cdot 7 + 9 \cdot 5 + 10 \cdot 10 \equiv$
$\equiv 0 \mod 11$
und somit gilt für die Differenz

$$\begin{aligned} 2 \cdot 5 - 2 \cdot 6 + 6 \cdot 6 - 6 \cdot 5 &\equiv 0 \mod 11 \\ 2 \cdot (5 - 6) + 6 \cdot (6 - 5) &\equiv 0 \mod 11 \\ 2 \cdot (5 - 6) - 6 \cdot (5 - 6) &\equiv 0 \mod 11 \\ (2 - 6) \cdot (5 - 6) &\equiv 0 \mod 11. \end{aligned}$$

Da $\text{ggT}(2 - 6, 11) = 1$*, ergibt sich, dass* $5 - 6 = 0 \mod 11$*, also ein Widerspruch.*

Neben der Fehlererkennung (Prüfgleichung ist erfüllt) ist es auch möglich einzelne Fehler zu korrigieren. Leider vermag das Verfahren im Allgemeinen nicht aus einer ungültigen die gültige ISBN zu ermitteln. Dazu muss dazu die Stelle, an der der Fehler aufgetreten ist, bekannt sein. Es können also beispielsweise einzelne unkenntliche Ziffern wieder bestimmt werden, was in folgender Aufgabe erledigt werden soll.

Aufgaben 23. *Finde jeweils die Variablen (alle Möglichkeiten), sodass sich eine gültige ISBN ergibt.*

i) $342a551243$

ii) $342a55a243$

iii) $342a5512a3$

iv) $342ab51243$

v) $aaaaaaaaaa$

vi) $abababababab$

Aufgaben 24. *Könnte eine ISBN länger oder kürzer sein, ohne ihre fehlererkennenden Eigenschaften einzubüßen? Was kann man beim Verkürzen überdies feststellen?*

3 Weitere Prüfziffernverfahren

Um weitere leistungsstarke Prüfziffernverfahren beschreiben zu können, muss man sich zwei nicht-kommutativen Gruppen zuwenden, nämlich der symmetrischen Gruppe und der Diedergruppe.

3.1 Permutationen

Auch das Durchmischen von Zahlen kann man mathematisch fassen:
Dafür schreibt man kurz (einer Wertetabelle vergleichbar):

$$\sigma = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \end{pmatrix} \qquad \text{statt:} \qquad \begin{array}{c||c|c|c} x & 1 & 2 & 3 \\ \hline \sigma(x) & 1 & 3 & 2 \end{array}$$

Das bedeutet, dass die 1 auf 1, die 2 auf 3 und die 3 auf 2 geht.
Man nennt σ eine Permutation der Zahlen 1, 2, 3 und sagt, dass 2 auf 3 abgebildet wird. Dafür schreibt man $\sigma(2) = 3$.
Alle Möglichkeiten für Permutationen der Zahlen 1, 2 und 3:

$$\begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix}$$

Das entspricht auch den verschiedenen Arten die drei Zahlen in einer Reihe anzuordnen.

Insgesamt gibt es $3 \cdot 2 \cdot 1 = 3!$, also 6 Möglichkeiten.

Die Menge aller Permutationen bildet die Gruppe der Permutationen $\mathfrak{S}_3$.

Eine erste Anwendung der Permutationen kann durch die Vorstellung eines Prüfziffernverfahrens für Kontonummern erfolgen, das die Quersumme benutzt.

Beispiel 25. *Manche Banken multiplizieren beim Aufstellen ihrer Prüfgleichungen für die Kontonummern nicht nur, sondern benutzen die Quersumme (mit Q notiert). Eine Kontonummer 19645522 kann etwa folgende Prüfgleichung haben*

$$Q(2 \cdot a_8) + a_7 + Q(2 \cdot a_6) + a_5 + Q(2 \cdot a_4) + a_3 + Q(2 \cdot a_2) + a_1 \equiv 0 \mod 10.$$

Damit ergibt sich $Q(2) + 9 + Q(12) + 4 + Q(10) + 5 + Q(4) + 2 \equiv 0 \mod 10$, *also* $2 + 9 + 3 + 4 + 1 + 5 + 4 + 2 = 30 \equiv 0 \mod 10$.

Hierbei kann man $Q(2 \cdot x)$ auch als Permutation schreiben:

$$Q = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 0 & 2 & 4 & 6 & 8 & 1 & 3 & 5 & 7 & 9 \end{pmatrix}$$

Die Verkettung als ein Hintereinanderausführen von Funktionen ist aus dem Analysisunterricht bekannt und kann hier im Zusammenhang mit Permutationen gesehen werden, auch kann man hervorheben, dass die Verkettung nicht kommutativ ist.
Bemerkenswert ist überdies, dass man bei der Prüfgleichung für die Nummern auf den DM-Geldscheinen eine Permutation mehrfach hintereinander auszuführen hat.
Auch eine Verkettung von Permutationen ist möglich:
Wendet man etwa auf die 2 die Permutation Q an, so ergibt sich 4. Nun wendet man ein zweites Mal Q an und erhält 8. Dafür kann man auch $Q^2(2) = 8$ schreiben und es ergibt sich eine neue Permutation, wenn man das für alle Zahlen überlegt:

$$Q^2 = Q \circ Q = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 0 & 4 & 8 & 3 & 7 & 2 & 6 & 1 & 5 & 9 \end{pmatrix}$$

Mit $\sigma = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$ und $\tau = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix}$

ergibt sich $\sigma \circ \tau = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix} \circ \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \end{pmatrix}$,

aber $\tau \circ \sigma = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix} \circ \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix} \neq \sigma \circ \tau.$

Die Gruppe der Permutationen ist nicht kommutativ.

3.2 Die Diedergruppe

Neben der symmetrischen Gruppe soll noch die Diedergruppe als nicht-kommutative Gruppe vorgestellt werden.
Die Diedergruppe wird neben der Verkettung von Permutationen zum Aufstellen der Prüfgleichung für die DM-Geldscheine benötigt.
Ein regelmäßiges Fünfeck kann durch Spiegelung und Drehung auf 10 Arten auf sich selbst abgebildet werden (Deckabbildung).
Man kann mit den Deckabbildungen des Fünfecks „wie gewohnt“ rechnen, wenn man beachtet:
a entspricht einer Drehung um 72° nach links
b einer Spiegelung an einer senkrechten Achse.
Dabei wird stets mit der Grundstellung $a^0 = 1$ begonnen, um die verschiedenen Drehungen und Spiegelungen (der Reihe nach von rechts nach links) vorzunehmen. Es bedeutet aba^3 beispielsweise, dass das Fünfeck aus der Grundstellung 3 mal um 72° Grad nach links gedreht wird, anschießend gespiegelt, also umgedreht, und danach nochmals um 72° nach links gedreht wird.

Aufgabe 26.

i) Vereinfache $a^3b^2a^6ba^2a^4b^7$.

ii) Begründe $a^5 = a^0 = 1$ *und* $ab = ba^4$. *Können diese Regeln helfen Aufgabe (i) schneller (ohne Hilfe eines Fünfeckmodells) zu berechnen?*

Das Verfahren der 10-stelligen ISBN ist am effektivsten (Erkennen von Einzel- und allgemeine Drehfehlern). Es werden dazu 11 Ziffern benötige, da im Körper $\mathbb{Z}/11\mathbb{Z}$ gerechnet wird. Möchte man dagegen mit nur 10 Ziffern auskommen, muss man sich

mit dem Ring $\mathbb{Z}/10\mathbb{Z}$ zufrieden geben und es können weniger Fehler erkannt werden (entweder Einzel- oder Drehfehler).
Das Verfahren der DM-Geldscheine mit der Diedergruppe und der Verwendung von Permutationen stellt eine Art Kompromiss dar: Man benötigt nur 10 Zeichen, aber es werden sämtliche Einzel- und Nachbarschaftsdrehfehler erkannt.

4 Allgemeine Definition von Codes

Nachdem bis jetzt stets von Prüfgleichungen gesprochen wurde, soll nun allgemeiner und mathematisch exakt erklärt werden, was man unter einem Code versteht.
Eine Prüfgleichung sortiert „gute“ (die Gleichung ist erfüllt) von „schlechten“ (die Gleichung ist nicht erfüllt) aus.

Beispiel 27. *Für (c_1, c_2) mit $c_1, c_2 \in \mathbb{Z}/5\mathbb{Z}$ können 25 mögliche Codewörter in Frage kommen. Die Prüfgleichung*

$$c_1 + 3 \cdot c_2 \equiv 0 \mod 5$$

wird aber nur von 5 Belegungen erfüllt. Die dadurch festgelegte Menge

$$C = \{00, 13, 21, 34, 42\}$$

nennt man Code.

4.1 Codes

Definition 28. *Sei $A = \{a_1, a_2, \cdots, a_n\}$ eine endliche Menge, die man in diesem Zusammenhang Alphabet nennt. Ein Wort der Länge k besteht aus k Buchstaben — man sagt auch k-Tupel.*

Beispiel 29. $A = \{0, 1, a, ?, \subset, \aleph\}$

θ leeres Wort

1

?

?ℵ1 *man könnte dafür auch* $(?, \aleph, 1)$ *schreiben*

aa ⊂?ℵ?111 *sind Wörter*

wort ist kein Wort

Definition 30. *Sei A ein Alphabet. Mit A^* bezeichnet man die Menge aller Wörter über A. Für ein Wort $x \in A^*$ bezeichnet man mit* $\text{len}(x)$ *dessen Länge (Anzahl an Buchstaben).*

Beispiel 31. $A = \{0, 1, a, ?, \subset, \aleph\}$

$\text{len}(\theta) = 0$

$\text{len}(0) = 1$

$\text{len}(11) = 2$

$\text{len}(aa \subset ?\aleph?111) = 9$

Definition 32.

$$A^n := \{x \in A^* | \text{len}(x) = n\}$$

$$A_n := \{x \in A^* | \text{len}(x) \leq n\}$$

Bemerkung 33.

$$A^n \subset A_n \subset A^*$$

dabei gilt: $|A^n| = |A|^n$ *und* $|A_n| = 1 + |A| + |A|^2 + \cdots + |A|^n = \frac{|A|^{n+1}-1}{|A|-1}$ *(geometrische Reihe).*

Definition 34. *Sei A ein Alphabet mit r Buchstaben. Ein Code C ist eine Teilmenge von A^*:*

$$C \subset A^*.$$

Ist A zweielementig (meist $A = \mathbb{Z}/2\mathbb{Z}$), spricht man von einem binären Code; ist A dreielementig ($\mathbb{Z}/3\mathbb{Z}$), von einem tertiären Code. Im allgemeinen Fall mit $A = \mathbb{Z}/r\mathbb{Z}$ handelt es sich um einen r-ären Code.
Die Anzahl der Elemente von C nennt man auch Größe M des Codes C. Es gilt also: $|C| = M$.

Beispiel 35. *Mit $A = \mathbb{Z}/2\mathbb{Z}$ ist $C = \{0, 10, 110, 110\}$ ein binärer Code.*

VORSICHT: $c \in C$ ist keine Zahl in einem anderen Stellenwertsystem.
Ziel der Codierungstheorie ist es ja Informationen sicher zu halten. Dazu muss die zu schützende Information (ein Text der Umgangssprache oder ein Computerprogramm aus 0 und 1) durch die zur Verfügung stehenden Codewörter ausgedrückt werden.
Dabei spielt der Kommunikationskanal mit seinen verschiedenen Übertragungswahrscheinlichkeiten eine wichtige Rolle.
Zuordnung von Buchstaben oder auch Wörtern (der Umgangssprache) zu Codewörtern (der Mathematik)

a $\to$ 0 c $\to$ 110

b $\to$ 10 d $\to$ 1110

Codierung: acd $\to$ 01101110

Decodierung: 1110100 $\to$ dba

Alle Codewörter müssen nicht die gleiche Länge haben, um eine eindeutige Decodierung zu sichern.

Beispiel 36.

i) $A = \{0, 1, 2, \cdots 9\}$
$C = \{00, 01, 02, 03, 04 \cdots 25\}$ *mit*
$a \to 00, \quad b \to 01, \; c \to 02, \quad d \to 03, \; e \to 04, \; \cdots \quad z \to 25$
ist eindeutig decodierbar.

ii) $A = \{0, 1, 2, \cdots 9\}$
$C = \{0, 1, 2, \cdots 10, 11 \cdots 25\}$ *mit*
$a \to 0, \quad b \to 1, \; c \to 2, \quad \cdots \; l \to 11, \quad m \to 12, \; \cdots \quad z \to 25$
ist nicht eindeutig decodierbar, da $\left.\begin{array}{r} bat \\ babj \\ kt \end{array}\right\} \to 1019.$

iii) ASCII-Code (vgl. Blatt) ist eindeutig decodierbar; alle Codewörter haben die gleiche Länge.

Die Decodierung ist bei einem Code mit fester Länge besonders einfach, weshalb diese Gruppe einen eigenen Namen erhält.

Definition 37. *Ein Block-Code oder Code fester Länge besteht nur aus Wörtern einer bestimmten Länge n. Man nennt n auch Länge des Codes.*

Bemerkung 38. *Block-Codes haben Vor- und Nachteile*

- \+ *keine Trennzeichen nötig und leicht decodierbar*
- − *häufige und weniger häufige Zeichen benötigen den gleichen Platz (→ Verschwendung von Speicherplatz)*

4.2 Hamming-Abstand

Definition 39. *Sind x und y zwei Codewörter gleicher Länge, so bezeichnet der Hamming-Abstand*

$$\mathrm{d}(x, y)$$

die Anzahl der Stellen, an denen sich x und y unterscheiden.

Beispiel 40.

i) $\mathrm{d}(387947043, 387497043) = 2$

ii) $\mathrm{d}(\mathrm{tot}, \mathrm{rot}) = 1$

iii) $\mathrm{d}(1234567, 7654321) = 6$

iv) $\mathrm{d}(00000, 11111) = 5$

Beispiel 41. *Man kann den Hamming-Abstand zur Korrektur von Fehlern verwenden:*
$C = \{0000, 0011, 1000, 1100\}$
empfangen: $x = 0111$

$$\left.\begin{array}{l} \mathrm{d}(0000, 0111) = 3 \\ \mathrm{d}(0011, 0111) = 1 \\ \mathrm{d}(1000, 0111) = 4 \\ \mathrm{d}(1100, 0111) = 3 \end{array}\right\} \quad \rightarrow \qquad \text{Decodierung} : c = 0011$$

Das Wort mit dem geringsten Abstand dient zur Decodierung, bei mehreren Worten mit dem gleichen Abstand hat man (leider) freie Wahl.

Satz 42 (Eigenschaften des Hamming-Abstands (einer Metrik))**.**
Seien $x, y, z \in A^n$.

i) $\mathrm{d}(x,y) \geq 0$
$\mathrm{d}(x,y) = 0 \quad \Leftrightarrow \quad x = y$

ii) $\mathrm{d}(x,y) = \mathrm{d}(y,x)$

iii) $\mathrm{d}(x,z) \leq \mathrm{d}(x,y) + \mathrm{d}(y,z)$

4.3 Fehlererkennung und -korrektur

Definition 43. *Ein Code C, dessen Codewörter alle feste Länge n haben, heißt u-fehlererkennend, falls ein Wort, das aus einem Codewort durch mindestens einen, aber maximal u Fehler entsteht, kein Codewort ist.*

Beispiel 44. $C = \{000000, 111000, 111111\}$ *ist 2-fehlererkennend.*

Definition 45. *Ein Code C, dessen Codewörter alle feste Länge n haben, heißt v-fehlerkorrigierend, falls die Korrektur mit dem Hamming-Abstand v oder weniger Fehler korrigiert (vgl. Beispiel 41).*
D.h. wurde ein Wort x empfangen, so sucht man ein $c \in C$ mit

$$\mathrm{d}(c,x) = \min\{\mathrm{d}(c',x) | c' \in C\}.$$

Beispiel 46. $C = \{0000000000, 1111100000, 1111111111\}$ *ist 2-fehlerkorrigierend.*

Beispiel 47. $\mathrm{Rep}_2(3) = \{000, 111\}$ *ist* 2*-fehlererkennend und* 1*-fehlerkorrigierend.*
$\mathrm{Rep}_r(n) = \{00\cdots0, 11\cdots1, \cdots, (r-1)(r-1)\cdots(r-1)\}$ *der r-äre Wiederholungscode der Länge n ist $n-1$-fehlererkennend und $\left[\frac{n-1}{2}\right]$-fehlerkorrigierend.*

Bemerkung 48. *Ist ein Code v-fehlerkorrigierend, so ist er auch v-fehlererkennend.*

4.4 Minimalabstand

Der kleinste Abstand zwischen zwei Codewörtern gibt auch Auskunft über die Fehlererkennung.

Definition 49. *Sei C ein Code, dessen Codewörter alle die feste Länge n haben.*

$$\mathrm{d}(C) := \min\{\mathrm{d}(c,d) | c,d \in C; c \neq d\}$$

heißt Minimalabstand des Codes C.

Es gilt stets: $\mathrm{d}(C) \geq 1$.

Beispiel 50.

i) *Für* $C_1 = \{000, 010, 111\}$ *ist* $\mathrm{d}(C_1) = 1$, *da* $\mathrm{d}(000, 010) = 1$.

ii) $C_2 = \{00011, 00101, 11101, 11000\}$

d	00011	00101	11101	11000
00011	/	2	4	4
00101	2	/	2	4
11101	4	2	/	2
11000	4	4	2	/

also: $\mathrm{d}(C_2) = 2$

iii) $C_3 = \mathrm{Rep}_r(n)$, *also:* $\mathrm{d}(C_3) = n$

iv) $C_4 = \{0000000000, 1111100000, 1111111111\}$, *also:* $\mathrm{d}(C_4) = 5$

Satz 51. *Sei C ein Code, dessen Codewörter alle eine feste Länge n haben.*

i) *C ist genau dann u-fehlererkennend, wenn* $\mathrm{d}(C) \geq u + 1$.

ii) *C ist genau dann v-fehlerkorrigierend, wenn* $\mathrm{d}(C) \geq 2v + 1$.

Beweis:

i) **1. Teil:** $\mathrm{d}(C) \geq u+1 \quad \Rightarrow \quad C$ ist u-fehlererkennend

Angenommen, c wird mit 1 bis u Fehlern übertragen und das dadurch empfangene Wort sei x, dann gilt:

$$1 \leq \mathrm{d}(c,x) \leq u.$$

Somit kann x keine Codewort sein, da $\mathrm{d}(C) \geq u+1$. C ist also u-fehlererkennend.

2. Teil: C ist u-fehlererkennend $\quad \Rightarrow \quad \mathrm{d}(C) \geq u+1$

Angenommen, C ist u-fehlererkennend, aber $\mathrm{d}(C) \leq u$, dann gibt es $c, d \in C$ mit $\mathrm{d}(c,d) = \mathrm{d}(C) \leq u$.

Nun könnte d aus c bei der Übertragung entstanden sein, wobei höchstens u Fehler aufgetreten sind. Somit wird der Fehler aber nicht erkannt, da $d \in C$. Also kann C nicht u-fehlererkennend sein.

Das zeigt $\mathrm{d}(C) \geq u+1$.

ii) **1. Teil:** $\mathrm{d}(C) \geq 2v+1 \quad \Rightarrow \quad C$ ist v-fehlerkorrigierend

Angenommen, c wird mit 1 bis v Fehlern übertragen und das daraus entstandene Wort x wird empfangen, so gilt

$$1 \leq \mathrm{d}(c,x) \leq v.$$

Wäre C nicht v-fehlerkorrigierend, so gäbe es ein anderes Codewort $d \in C$, das näher oder gleich weit von x enfernt ist, also: $\mathrm{d}(x,d) \leq v$.

Somit ist $\mathrm{d}(c,d) \leq \mathrm{d}(c,x) + \mathrm{d}(x,d) \leq v + v = 2v$, was im Widerspruch zu $\mathrm{d}(C) \geq 2v+1$ steht. Also muss die Decodierung mit dem Hamming-Abstand c liefern und der Code ist v-fehlerkorrigierend.

2. Teil: C ist v-fehlerkorrigierend $\quad \Rightarrow \quad \mathrm{d}(C) \geq 2v+1$

Angenommen, C ist v-fehlerkorrigierend und $\mathrm{d}(C) \leq 2v$.

Es gibt dann $c, d \in C$ mit $\mathrm{d}(c,d) = \mathrm{d}(C) \leq 2v$.

Außerdem muss nach (i) gelten: $\mathrm{d}(c,d) \geq v+1$, da C nach Bemerkung 48 jedenfalls v-fehlererkennend ist.

$k := \mathrm{d}(C) \quad \Rightarrow \quad v+1 \leq k \leq 2v \quad \Rightarrow \quad 1 \leq k-v \leq v.$

c stimmt mit d in $n-k$ Stellen (z.B. den letzten) überein, wobei n die Codewortlänge ist.

Nun konstruiert man ein Wort x aus dem Codewort c.

Von den vorderen k Stellen von c ändert man v Buchstaben (z.B. die mittleren) durch entsprechende Buchstaben aus d ab und behält $k-v$ Stellen (z.B. die ersten) bei.

$$x = \underbrace{x_1 \cdots x_{k-v}}_{\text{nur } c} - \underbrace{x_{k-v+1} \cdots x_k}_{\text{nur } d} - \underbrace{x_{k+1} \cdots x_n}_{c \text{ und } d \text{ stimmen überein}}$$

Somit ergibt sich $\mathrm{d}(c,x) = v$ und $\mathrm{d}(d,x) = k - v \leq v$.

Also ist ein Wort x gefunden, das nicht eindeutig mit c decodiert wird, obwohl x aus c durch v Fehler entstanden ist.

□

Definition 52. *Sei C ein Code der Größe M (vgl. Definition 34), dessen Codewörter alle eine feste Länge n haben, mit Minimalabstand d. Man nennt C dann einen (n, M, d)-Code.*

Beispiel 53.

i) $C_1 = \{000, 010, 011\}$ *ein* $(3,3,1)$-*Code.*

ii) $C_2 = \{00011, 00101, 11101, 11000\}$ *ein* $(5,4,2)$-*Code.*

iii) $C_3 = \{0000, 1100\}$ *ein* $(4,2,2)$-*Code.*

iv) $C_4 = \{00, 01, 10, 11\}$ *ein* $(2,4,1)$-*Code.*

v) $\mathrm{Rep}_r(n)$ *ein* (n, r, n)-*Code.*

5 Lineare Codes

Beispiel 54. *Löse das Gleichungssystem über* $\mathbb{Z}/2\mathbb{Z}$

I) $c_1 + c_2 + c_3 = 1$

II) $c_2 - c_3 = 0 \quad \Rightarrow \quad c_2 = c_3$

III) $c_1 + c_2 = 1$

Somit ergibt sich:

I') $c_1 + 2c_3 = 1$

III') $c_1 + c_3 = 1 \quad \Rightarrow \quad c_1 = 1 - c_3$

Also ist I") $1 - c_3 + 2c_3 = 1 \quad \Rightarrow \quad c_3 = 0$ *und somit* $c_2 = 0$ *und* $c_1 = 1$*, was zu* $\mathbb{L} = \{(1, 0, 0)\}$ *führt.*

Definition 55. *Ein Gleichungssystem, in dem nur Vielfache der Variablen vorkommen (keine Konstanten), nennt man ein homogenes lineares Gleichungssystem.*

Beispiel 56. *Löse das homogene lineare Gleichungssystem über* $\mathbb{Z}/2\mathbb{Z}$

I) $c_1 + c_2 + c_3 = 0$

II) $c_2 + c_3 = 0 \quad \Rightarrow \quad c_3 = -c_2 = -c_2 + 0 = -c_2 + 2c_2 = c_2$

III) $c_2 + c_4 = 0 \quad \Rightarrow \quad c_4 = -c_2 = c_2$

Durch Addition von I) und II) ergibt sich unter Beachtung von $1 + 1 = 0$*:*

$I + II)$ $c_1 = 0$.

Also ist $\mathbb{L} = \{(0, c, c, c) | c \in \mathbb{Z}/2\mathbb{Z}\} = \{0000, 0111\}$.

Definition 57. *Durch die Lösungsmenge von Gleichungssystemen können Codes angegeben werden.*

Bei solchen Codes, die sich aus homogenen linearen Gleichungssystemen über $\mathbb{Z}/p\mathbb{Z}$ *ergeben, spricht man von linearen Codes.*

Ein homogenes lineares Gleichungssystem kann auch in einer Kurzform geschrieben werden, falls man es in Form einer Matrix schreibt.

Diese Matrix nennt man im Zusammenhang mit Codes **Prüfmatrix**. Man kann damit prüfen, ob ein Wort zum Code gehört. Dazu muss dieses Wort alle Gleichungen des Systems erfüllen.

Beispiel 58.

i) Die Prüfmatrix über $\mathbb{Z}/2\mathbb{Z}$

$$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

stellt das lineare Gleichungssystem über $\mathbb{Z}/2\mathbb{Z}$ *dar:*

I) $c_1 + c_2 = 0 \quad \Rightarrow \quad c_1 = -c_2 = c_2$

II) $c_2 + c_3 = 0 \quad \Rightarrow \quad c_2 = -c_3 = c_3$

III) $c_3 + c_5 = 0 \quad \Rightarrow \quad c_3 = -c_5 = c_5$

Somit erhält man $c_1 = c_2 = c_3 = c_5$ *und* $c_4 = c_4$*, also*

$\mathbb{L} = \{(c, c, c, d, c) | c, d \in \mathbb{Z}/2\mathbb{Z}\} = \{00000, 11101, 00010, 11111\}$.

ii) Die Prüfmatrix über $\mathbb{Z}/2\mathbb{Z}$ *zum Gleichungssystem aus Beispiel 56 ist*

$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}.$$

iii) Die Prüfmatrix für die ISBN ist $\begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{pmatrix}$*, da sie auf die Prüfgleichung* $a_1 + 2a_2 + 3a_3 + \cdots 10a_{10} = 0$ *in* $\mathbb{Z}/11\mathbb{Z}$ *führt.*

Hierbei hat man aber die Besonderheit nicht berücksichtigt, dass die $10 = X$ *nur an der letzten Stelle erscheinen darf.*

Bemerkung 59. *Man kann die Wörter eines linearen Codes* C *komponentenweise addieren und ebenfalls komponentenweise mit einem Buchstaben multiplizieren. Dabei ergibt sich wieder ein Codewort aus* C*. Man spricht in diesem Zusammenhang von der Abgeschlossenheit eines linearen Codes.*

Man sagt auch: Ein linearer Code ist ein Vektorraum (ein Untervektorraum von $(\mathbb{Z}/p\mathbb{Z})^n$*).*

Beim Code $C = \{00000, 11101, 00010, 11111\}$, der sich aus Beispiel 58 (i) ergibt, gilt etwa

$11101 + 11111 = 00010$ oder $1 \cdot 11101 = 11101$,

wenn man bedenkt man, dass man zwei Codewörter addiert, indem man die Buchstaben jeder Komponente addiert, oder mit einem Buchstaben multipliziert, indem man die Buchstaben jeder Komponente mit diesem multipliziert.

Somit kann C auch noch durch eine anderes Verfahren erzeugt werden:

Findet man Codewörter, aus denen alle übrigen Codewörter durch diese beiden Möglichkeiten berechnet werden können, sagt man, dass dann diese Codewörter den gesamten Code erzeugen.

Eine minimale Anzahl solcher Codewörter nennt man eine Basis des Codes.

Im vorliegenden Beispiel sind 00010 und 11101 eine Basis, da

$0 \cdot 00010 = 00000$ und $00010 + 11101 = 11111$ und eines der Wörter offensichtlich nicht zur Erzeugung des gesamten Codes reicht.

Die Wörter der Basis kann man wiederum in Form einer Matrix schreiben, indem man jedes Codewort der Basis in eine Zeile schreibt. Diese Matrix nennt man nun **Erzeugermatrix**:

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

Es kann für einen Code verschiedene Basen und somit auch verschiedene Erzeugermatrizen geben.

Definition 60. *Einen linearen Code der Länge n mit Minimalabstand d, der durch ein homogenes lineares Gleichungssystem mit n Variablen festgelegt werden kann, das in der Lösungsmenge noch k Variablen frei wählen lässt, nennt man einen $[n, k, d]$-Code.*

oder: *Einen linearen Code der Länge n mit Minimalabstand d, der durch eine Erzeugermatrix mit n Spalten und k Zeilen festgelegt werden kann, nennt man einen $[n, k, d]$-Code.*

Bemerkung 61. *Jeder lineare r-äre $[n, k, d]$-Code ist ein (n, r^k, d)-Code.*

Beispiel 62.

i) Bestimme eine Erzeugermatrix des binären linearen Codes C_1, der durch folgende Prüfmatrix gegeben ist:

$$\begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

I) $\quad c_1 + c_4 = 0 \quad \Rightarrow \quad c_1 = c_4$

II) $\quad c_1 + c_2 + c_3 = 0 \quad \Rightarrow \quad c_2 = c_1 + c_3 = c_3 + c_4$

Also ist $C_1 = \{(c, c+d, d, c) | c, d \in \mathbb{Z}/2\mathbb{Z}\} = \{0000, 1101, 0110, 1011\}$ *ein binärer* $[4, 2, 2]$*- oder* $(4, 4, 2)$*-Code.*

Dieser Code kann durch die Wörter 1011 *und* 0110 *erzeugt werden, da insbesondere* $1011 + 0110 = 1101$*. Somit ergibt sich als Erzeugermatrix:*

$$\begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

Da insbesondere $1101 + 1011 = 0110$*, ergibt sich als eine andere Erzeugermatrix:*

$$\begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}.$$

ii) Bestimme den tertiären linearen Code C_2, der durch folgende Erzeugermatrix gegeben ist:

$$\begin{pmatrix} 0 & 1 & 2 & 1 \\ 2 & 2 & 1 & 0 \end{pmatrix}.$$

$+$		$0 \cdot 0121$	$1 \cdot 0121$	$2 \cdot 0121$
$0 \cdot 2210$	$C_2 =$	$\{0000,$	$0121,$	$0212,$
$1 \cdot 2210$		$2210,$	$2001,$	$2122,$
$2 \cdot 2210$		$1120,$	$1211,$	$1002\}$

ist ein tertiärer $[4, 2, 2]$*- oder* $(4, 9, 2)$*-Code.*

iii) Bestimme den binären linearen Code C_3 mit der Basis 11001 *und* 01101*.*

$C_3 = \{00000, 11001, 01101, 10100\}$ *ein binärer* $[5, 2, 2]$*- oder* $(5, 4, 2)$*-Code.*

Definition 63. *Das Gewicht eines Codewortes c ist die Anzahl der Stellen, die nicht Null sind und wird mit* $\mathrm{w}(c)$ *bezeichnet.*
Unter dem Gewicht $\mathrm{w}(C)$ *eines Codes C versteht man das Minimum aller Gewichte der Codewörter, die nicht vollständig Null sind.*

Beispiel 64. *Für die Codes aus Beispiel 62 gilt:*

i) $\mathrm{w}(1101) = 3$, $\mathrm{w}(0110) = 2$, $\mathrm{w}(1011) = 3$
also ist $\mathrm{w}(C_1) = 2 = \mathrm{d}(C_1)$.

ii) $\mathrm{w}(C_2) = 2 = \mathrm{d}(C_2)$

iii) $\mathrm{w}(C_3) = 2 = \mathrm{d}(C_3)$

Das Beispiel legt folgende Aussage nahe:

Bemerkung 65. *Für einen linearen Code C gilt*

$$\mathrm{d}(C) = \mathrm{w}(C).$$

6 Decodierung bei linearen Codes

Da auf Grund der Größe M linearer Codes die Decodierung nach dem oben beschriebenen Muster (vgl. Beispiel 41 und Definition 45) mit dem Hamming-Abstand sehr aufwändig wird, legt man sich eine kürzere gleichwertige Methode zurecht.
Die Decodierung soll am Beispiel eines binären, linearen Codes

$$C = \{0000, 1011, 0110, 1101\}$$

mit der Prüfmatrix

$$P = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

gezeigt werden.
Um Verwechslungen zu vermeiden und zur Erinnerung:

$$E = \begin{pmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$$

war die Erzeugermatrix.

Verfahren 66 (Standardtabelle)**.** *Man erstellt aus den Worten eines p-ären linearen (n, M, d)-Codes C nach folgendem Muster eine Tabelle, in der alle Wörter aus $(\mathbb{Z}/p\mathbb{Z})^n$ enthalten sind:*

i) In der Kopfleiste werden alle Codewörter angetragen, wobei $c_0 = 00\cdots0$ stets das erste ist.

ii) Nun wird ein Wort f_2, das nicht in C, aber in $(\mathbb{Z}/p\mathbb{Z})^n$ enthalten ist, gewählt, das unter allen verbliebenen Wörtern minimalen Abstand hat. In dieser Reihe werden die Wörter eingetragen, die sich durch (komponentenweise) Addition von f_2 und allen Codewörtern ergeben.

iii) Sodann wird ein Wort f_3 aus $(\mathbb{Z}/p\mathbb{Z})^n$, das noch nicht in der Tabelle steht, gewählt, das unter allen diesen verbliebenen Wörtern minimalen Abstand hat. In dieser Reihe werden wieder die Wörter eingetragen, die sich nun durch Addition von f_3 und allen Codewörtern ergeben.

iv) Dieses Verfahren wird solange durchgeführt bis alle Wörter aus der Menge

$$(\mathbb{Z}/p\mathbb{Z})^n$$

aufgebraucht sind.

Es ergibt sich damit folgende Tabelle:

$c_0 = 0$	c_1	c_2	$\cdots$	c_M
f_2	$c_1 + f_2$	$c_2 + f_2$	$\cdots$	$c_M + f_2$
f_3	$c_1 + f_3$	$c_2 + f_3$	$\cdots$	$c_M + f_3$
$\vdots$	$\vdots$	$\vdots$		$\vdots$
f_q	$c_1 + f_q$	$c_2 + f_q$	$\cdots$	$c_M + f_q$

Hierbei ist $M = p^k$ und $|(\mathbb{Z}/p\mathbb{Z})^n| = p^n$, also ergibt sich $q = \frac{p^n}{M} = \frac{p^n}{p^k} = p^{n-k}$.

Beispiel 67. *Für das Beispiel des Codes C ergibt sich etwa:*

0000	1011	0110	1101
1000	0011	1110	0101
0100	1111	0010	1001
0001	1010	0111	1100

Verfahren 68 (Decodierung)**.** *Soll nun ein empfangenes Wort x decodiert werden, so sucht man dieses Wort in der Tabelle und decodiert es mit dem Codewort, das in der Spalte ganz oben steht.*

Beispiel 69. *Wird beispielsweise $x = 0010$ empfangen, so wird dieses Wort mit dem Codewort $c_2 = 0110$ decodiert. Das Wort am Anfang der Zeile nennt man Fehlerwort (oder Nebenklassenführer) $f_3 = 0100$, dann gilt:*

$$x = c_2 + f_3 \qquad \text{oder} \qquad c_2 = x - f_3.$$

Dieses Verfahren ist aufwändig, da jedes Wort explizit in die Tabelle eingetragen werden muss. Zusätzlich ist das Erstellen der Tabelle nicht eindeutig, da es beim Ausfüllen der ersten Spalte der Tabelle manchmal mehrere Wörter zur Auswahl gibt, und somit ist die Decodierung in gewissen Fällen abhängig von der Wahl der Fehlerwörter in der Tabelle.
Bei der Decodierung mit Verfahren 68 können Fehler korrigiert werden. Dies entspricht dem Verfahren nach Definition 45 aus Beispiel 41. Lediglich in den Fällen bei denen nach Definition 45 keine eindeutige Entscheidung getroffen werden kann und man zu wählen hat, wie decodiert werden soll, ergibt sich mit Verfahren 68 ein eindeutiges Ergebnis. Dies liegt aber daran, dass hier schon eine Entscheidung bei der Wahl der Fehlerwörter getroffen werden musste.

Bemerkung 70. *Die Decodierung mit dem Minimalabstand (vgl. Beispiel 41) ist nicht eindeutig*
$\mathrm{d}(0000, 0010) = 1$
$\mathrm{d}(0110, 0010) = 1.$
Mit obiger Tabelle ist die Decodierung eindeutig, da man sich bei der Wahl der f_i bereits entscheiden musste.

Hinweis 71. *Man schreibt die Zeilen einer Matrix* $P = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$ *als Spalten und erhält* $P^t = \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}$, *die transponierte Matrix genannt wird.*
Für ein Worte $x = 0111$ *und die Matrix* P^t *berechnet sich das Produkt* $x \cdot P^t$ *nach folgendem Muster:*

$$x \cdot P^t = 0111 \cdot \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} =$$

$$= 0 \cdot 1 + 1 \cdot 0 + 1 \cdot 0 + 1 \cdot 1 \qquad 0 \cdot 1 + 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 0 = 10$$

Verfahren 72 (Syndromtabelle)**.** *Das Syndrom $S(x)$ eines Wortes $x = 0111$ berechnet sich als Produkt $x \cdot P^t$ nach Hinweis 71:*

$$S(x) = x \cdot P^t.$$

Auf diese Weise erhält man aus der Standardtabelle die folgende Syndromtabelle.

Nebenklassenführer	*Syndrom*
$f_1 = c_0 = 0$	$S(0) = 0$
f_2	$S(f_2)$
f_3	$S(f_3)$
$\vdots$	$\vdots$
f_q	$S(f_q)$

Beispiel 73. *Für das Beispiel des Codes C muss zunächst für jedes Codewort das Syndrom berechnet werden.*

$$0000 \cdot \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} = 00; \quad 1000 \cdot \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} = 11;$$

$$0100 \cdot \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} = 01; \quad 0001 \cdot \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} = 10$$

Somit erhält man folgende Syndromtabelle.

Nebenklassenführer	*Syndrom*
0000	00
1000	11
0100	01
0001	10

Verfahren 74 (Syndrom-Decodierung)**.** *Soll nun ein empfangenes Wort x decodiert werden, so muss nicht mehr in der Tabelle gesucht werden, sondern man berechnet das Syndrom und subtrahiert das entsprechende Fehlerwort.*

Beispiel 75. *Wird nun* $x = 0010$ *empfangen, so ist das Syndrom:*

$$0010 \cdot \begin{pmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} = 01.$$

Das zugehörige Fehlerwort ist $f_3 = 0100$ *und somit ergibt sich*

$$c = x - f_3 = 0010 - 0100 = 0110 = c_2.$$

Bemerkung 76. *Auch die Syndrom-Decodierung muss nicht eindeutig sein (vgl. Bemerkung 70).*

Verschiedene Nebenklassenführer haben verschiedene Syndrome.

Alle Wörter in einer Zeile einer Standardtabelle (vgl. Verfahren 66) haben das selbe Syndrom, wie der zugehörige Nebenklassenführer. Sie haben also alle das gleiche Fehlerwort.