



Arbeitsblatt in Informatik: „Prüfziffern“

Datum:

Name:

Kurs:

Aufgabe 1: ISBN-10

Auf (fast) jedem Buch findet man eine so genannte ISBN, eine eindeutige Nummer, um ein Buch zu identifizieren. Vertippt sich der Buchhändler oder der Bibliothekar bei einer Bestellung, kann das Ärger geben. Darum enthält die ISBN eine Prüfziffer:

Eine ISBN-10 ist 10stellig.

Die letzte Ziffer ist die Prüfziffer $P \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, X\}$, wobei X für die Ziffer 'Zehn' steht.

Beispiele: 3 528 16989 3 3 464 57315 X 3 594 10826 0

Alle Ziffern werden von rechts nach links, mit ihrer Position in der Ziffernfolge gewichtet, d.h. die Prüfziffer wird mit 1, die vorletzte Ziffer wird mit 2, die nächste mit 3, usw. multipliziert.

Der ISBN-Code ist so ausgelegt, dass die entstehende Summe plus Prüfziffer ohne Rest durch 11 teilbar ist. Beispiel: ISBN = 3 528 16989 3

$$10 \cdot 3 + 9 \cdot 5 + 8 \cdot 2 + 7 \cdot 8 + 6 \cdot 1 + 5 \cdot 6 + 4 \cdot 9 + 3 \cdot 8 + 2 \cdot 9 + 3 = 264 = 24 \cdot 11 \text{ (ohne Rest)}$$

Damit ist die ISBN gültig.

a) Ergänze die Prüfziffer für folgende ISBN-Nummern

- Fermat's Last Theorem. **ISBN:** 1 85702 669 ___
- The Code Book. **ISBN:** 0 385 49532 ___
- Big Bang: The Origin Of The Universe. **ISBN:** 0 007 16220 ___

b) Prüfe, ob folgende Nummern korrekt eingegeben wurden, falls nein, was könnte der Fehler sein?

- Lambacher Schweizer Klasse 9: **ISBN: 3 127 31059 1**
- Lambacher Schweizer GK Analysis: **ISBN: 3 127 32170 8**
- Gödel, Escher, Bach: **ISBN: 3 423 30017 5**
- Eine kurze Geschichte der Zeit: **ISBN: 3 499 50555 4**
- Per Anhalter durch die Galaxis: **ISBN: 3 453 14694 2**

Aufgabe 2: EAN-Codes

a) Was sind EAN-Codes? Erkläre den Aufbau eines EAN-13 Codes!

Die Prüfziffer der EAN, die letzte Ziffer (x_n), errechnet sich, indem die einzelnen Ziffern von rechts nach links, beginnend mit der vorletzten (x_{n-1}), abwechselnd mit 3 und 1 multipliziert und anschließend diese Produkte addiert werden. ($x_{n-1} \cdot 3 + x_{n-2} \cdot 1 + \dots$). Die Prüfziffer ergänzt diese Summe dann zum nächsten Vielfachen von 10.

Beispiel (Vanilla Coke), EAN: 544900009624-1

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

$$= 12 + 2 + 18 + 9 + 0 + 0 + 0 + 0 + 27 + 4 + 12 + 5 = 89$$

Die Prüfziffer muss also 1 sein, damit die insgesamt eine durch 10 teilbare Zahl herauskommt.



b) Berechne die Prüfziffer für folgende EAN-Nummern:

- 237542009624 ___
- 012345678910 ___
- 471108152342 ___

Aufgabe 3: Recherchiere, was eine „Data Matrix“ ist. Welche Vor- und Nachteile hat sie. Wo wird sie eingesetzt?



Arbeitsblatt in Informatik: „Prüfziffern“

Datum:

Name:

Kurs:

Aufgabe 1: Paritätsbit

Zur Erkennung von 1-Bit-Fehlern (also, wenn bei einer Datenübertragung nur ein Bit fehlerhaft ist) genügt ein so genanntes *Paritätsbit*.

Definition (Meyers Lexikon): Paritätsbit, Prüfbit : eine zusätzliche binäre Stelle (Bit) in einem binär codierten Zeichen oder Wort, das keine Information trägt, sondern der Fehlerkontrolle dient.

- Was versteht man unter gerader / ungerader Parität?
- An welcher Stelle kann das Paritätsbit sinnvoll in unseren *Rahmen* eingefügt werden?
- Beschreibe aufbauend auf unser Unterrichtsbeispiel ein erweitertes Protokoll für Sender- und Empfängerseite anhand eines selbst gewählten Beispiels. Benutze dabei eine „gerade Parität“.

Aufgabe 2: Prüfsumme

- Erläutere, wie sich das Protokoll aus Aufgabe 4 verändert, wenn man statt eines Paritätsbit die binäre Quersumme zur Prüfung speichert.

Man kann auch mehrere *Rahmen (Pakete)* zusammenfassen und für mehrere Byte eine Prüfsumme festlegen. Um dies nicht unnötig kompliziert zu machen gehen wir von der Dezimaldarstellung der Bytes aus (also Zahlen im Bereich 0-255).

Beispiel:

3-byte-Nachricht: 7,24,11 Prüfsumme: $7+24+11 = 42$ übertragen: 7,24,11,42

Empfänger rechnet und vergleicht: $7+24+11 = 42$, $42 = 42$, alles in Ordnung!

Fehlerfälle: $7+23+11 = 41 \neq 42$, Fehler erkannt!

$6+24+12 = 42 = 42$, Fehler nicht erkannt!

$24+7+11 = 42 = 42$, Fehler nicht erkannt!

- Kategorisiere die drei Beispielfehler. Gib für jeden Fehlertyp einen weiteren Fehler an.

Aufgabe 3: „Prüfreste“

Das Prüfsummen-Verfahren aus Aufgabe 5 ist recht leicht auszuhebeln.

Ein "Prüfquotienten-Verfahren":

Nachricht: 7,24,11. Aus den Bytes wird eine 9-stellige Zahl gebildet: 007024011, die Zahl wird durch einen festen "geeigneten" Divisor geteilt: $007024011 : 171 = 41\ 076$ Rest 15, der Rest wird als "Prüfzahl" benutzt.

Übertragen wird: 7,24,11,15.

- Warum wird für dieses Beispiel eine 9-stellige Zahl gewählt?
- Welche Zahlen eignen sich gut als Divisor d ? Wie ist es mit $d=2$, $d=100$, $d=1000$, oder $d=999999999$?
- Versuche Beispiele zu finden, bei denen das Verfahren scheitert. Bewerte das Verfahren anhand Deiner Beobachtungen.

Aufgabe 4: CRC – „cyclic redundancy check“

- Recherchiere, wie das CRC – Verfahren funktioniert und erläutere die wichtigsten Sachverhalte schriftlich anhand eines selbst gewählten Beispiels.
- Bluetooth benutzt CRC als Prüfverfahren mit folgendem *Generatorpolynom* : $x^5 + x^4 + x^2 + 1$, (also 110101) für die Daten-Übertragung. Ermittle die Prüfsumme für folgende Daten:

- 1011001011101011

- 1000000000000000

- 1111111111111111

- ...

Hilfen / geeignete Quellen:

http://de.wikipedia.org/wiki/Cyclic_Redundancy_Check

<http://www.flechtmann.net/crc>