

Grundkurs Informatik

Errata, 6. Auflage 2016

Hartmut Ernst, Jochen Schmidt und Gerd Beneken

Stand: 21. Mai 2023

S. 21, Bsp. 1.5: In der 2. Spalte der Umwandlung in eine Dualzahl muss es heißen:
 $158 : 2 = 79 \text{ Rest } 0$.

S. 21, Bsp. 1.6: Beim Ergebnis für den ganzzahligen Anteil fehlt ganz rechts eine 1. Es muss heißen:
 $39_{\text{dez}} = 100111_{\text{bin}}$
Die Rechnung selbst ist korrekt.

S. 41/42, Abschnitt „Das Alphabet des genetischen Codes“: im gesamten Abschnitt sind die Wörter Nukleinsäure(n) durch Aminosäure(n) zu ersetzen. (Danke an Frederik Büker für den Hinweis)

S. 57/58: Die Formel für Kombinationen mit Wiederholungen muss lauten:

$$C(m, n) = \binom{n+m-1}{m}$$

und entsprechend dann Beispiel 2.6d:

$$\binom{3+2-1}{2} = 6$$

(Danke an Philip Waritschlager für den Hinweis)

S. 93, Abb. 3.7: Im 4D-Hyperwürfel fehlt die gestrichelte Verbindung zwischen 1101 und 1001.

S. 103: In Tabelle 3.18 muss das Generatorpolynom zu CRC-1 $x + 1$ lauten.

S. 125: Im Text zu Tabelle 3.23 muss es heißen: „Somit sind noch fünf Plätze für längere Zeichenketten frei.“
(Danke an Erik Reischl für den Hinweis)

S. 126: In Abb. 3.16 entfällt die letzte Zeile („Gib letztes Präfix P aus“) ersatzlos.
(Danke an Erik Reischl für den Hinweis)

S. 143ff.: Die in Abschnitt 4.1.2 gegebene Definition für Transpositions-Chiffren ist falsch. Bei einem Transpositions-Chiffre handelt es sich um eine Permutation der Zeichen des Klartextes, die Zeichen ändern also lediglich ihre Position.

Tatsächlich beschrieben ist ein affiner Substitutions-Chiffre. Die Enigma verwendet eine polyalphabetische Substitution. Jede Transposition ist durch eine solche polyalphabetische Substitution darstellbar.

S. 158, Abschnitt Diffie-Hellman – Sicherheit, 3. Absatz, 1. Satz muss heißen:
„Eine Primzahl p heißt *sicher*, wenn $p - 1$ einen großen Primteiler q besitzt; ...“.

S. 182, 5.2.4 unter „Benachbarte Terme“ muss heißen:
Betrachtet man den Ausdruck

$$(\neg a \wedge b \wedge c) \vee (\neg a \wedge b \wedge \neg c) = (\neg a \wedge b) \wedge (c \vee \neg c) = \neg a \wedge b \quad ,$$

so erkennt man, dass die beiden in Klammern gesetzten Terme benachbart sind. Durch Ausklammern von $\neg a \wedge b$ und unter Beachtung von $c \vee \neg c = 1$ folgt dann das Ergebnis $\neg a \wedge b$.
(Danke an Jonas Becker für den Hinweis)

S. 181, Abb. 5.11 (b) und (c): Das 2. e_1 muss jeweils e_2 sein. (Danke an Holger Hinrichs für den Hinweis)

S. 220, Tabelle 5.15, Befehl DBRA. Muss heißen:

DBRA D_n , Marke Dekrementiere D_n und verzweige zu Marke, wenn $D_n \geq 0$ ist
(Danke an Andreas Bohne-Lang für den Hinweis)

S. 223, in „Kopieren eines Datenblocks“ muss die Zeile DBRA D_2 , LOOP... ersetzt werden durch:

DBRA D_0 , LOOP * dekrementiere D_0 , Sprung nach LOOP wenn $D_0 \geq 0$
(Danke an Andreas Bohne-Lang für den Hinweis)

S. 224, in „Berechnung einer Prüfziffer“ muss der Kommentar zu DBRA D_0 , LOOP ersetzt werden durch:

dekrementiere D_0 , Sprung nach LOOP wenn $D_0 \geq 0$
(Danke an Andreas Bohne-Lang für den Hinweis)

Außerdem muss die folgende Zeile nach ORG \$1000 * Startadresse eingefügt werden:

SUBI #1, D_0 * D_0 für Sprungbefehl vorbereiten

S. 224, in „Arithmetisches Mittel“ ist die Zeile

MOVE.W (A0)+, D_1 * Ersten Wert laden und Adresse inkrementieren
ersatzlos zu streichen.

In der Zeile DBRA D_2 , LOOP... muss der Kommentar ersetzt werden durch:

dekrementiere D_2 , Sprung nach LOOP wenn $D_2 \geq 0$
(Danke an Andreas Bohne-Lang für den Hinweis)

S. 224, in „Berechnung der ganzzahligen Wurzel aus einer 16-Bit Zahl“ ist bei der Iterationsgleichung ein Operator falsch ($-$ statt $+$). Die Formel muss lauten:

$$x_{i+1} = x_i - \frac{x_i^2 - a}{2x_i} = \frac{1}{2} \left(x_i + \frac{a}{x_i} \right) \quad .$$

Der Programmcode ist korrekt.

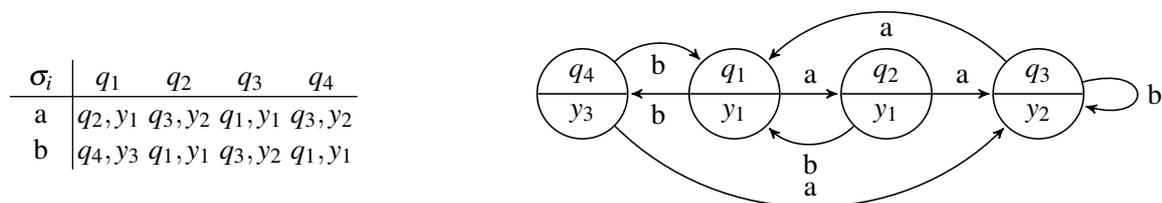
S. 253/254: Die Angabe zur maximalen Geschwindigkeit des ISA-Busses ist falsch. Es muss es heißen „Der ISA-Bus ist ein paralleler Bus mit einer Breite von 16 Bit, also 16 parallelen Datenleitungen. Bei einem Bustakt von 8 MHz wäre theoretisch eine maximale Datenübertragungsrate von knapp 16 MB/s möglich, die aber auf Grund von notwendigen Wartezyklen niemals erreicht werden kann.“ (Danke an Ryan Klarhölter für den Hinweis)

S. 322, Bsp. 8.4: In der letzten Zeile muss es heißen „... über `chmod go+r` jeweils das Leserecht eingeräumt.“ (Danke an Gaetano Paganini für den Hinweis)

S. 366, 2. Absatz, vorletzte Zeile von 9.8.3 muss heißen: „... einzig der Ort ist Pflichtfeld, `minoccurs=1`.“
(Danke an Gaetano Paganini für den Hinweis)

S. 374: Die Anzahl der möglichen Zustandsübergänge eines konkret gegebenen Automaten ist zn (DEA) bzw. zn^2 (NEA), wobei n die Anzahl der Zustände und z die Anzahl der Zeichen in Σ ist.

S. 375, Bsp. 10.3: Hier ist zwar ein Moore-Automat gezeigt, dieser ist aber anders als im Text beschrieben nicht äquivalent zum Mealy-Automat aus Bsp. 10.2. Ein äquivalenter Moore-Automat sieht so aus:



Jedem Zustand ist ein Ausgabezeichen zugeordnet, die Ausgabe ist nur noch vom Zielzustand abhängig. Zur Umformung des Mealy-Automaten aus Bsp. 10.2 musste der dortige Zustand q_2 in zwei Zustände q_2 und q_4 aufgeteilt werden, da die Ausgabe für a und b verschieden ist. Die Eingangspfeile wurde ebenfalls umgehängt, die Ausgangspfeile verdoppelt. (Danke an Enrico Winter für den Hinweis)

S. 376, Bsp. 10.4. Es muss heißen „Von q_1 ausgehend, gelangt man z. B. mit 10 zu q_5 .“ (Danke an Tilo Strutz für den Hinweis)

S. 382, 1. Zeile (Punkt 5) muss heißen: „Schreibe die in dieser Spalte neu aufgetretenen Teilmengen in die folgenden Spaltenköpfe.“ (Danke an Enrico Winter für den Hinweis)

S. 399, Bsp. 10.22: Im Text ist überall $a^4b^4c^4$ durch $a^4b^4a^4$ zu ersetzen. (Danke an Okan Karaoglan für den Hinweis)

S. 402 und S. 403, jeweils Pumping-Theorem: Das Wort w muss mindestens so lange wie n sein, also $|w| \geq n$. (Danke an Marc Wenninger für den Hinweis)

S. 406, Bsp. 10.27: In Schritt 3 muss es genauer heißen: „Regeln, bei denen ein Nichtterminal in ein einzelnes anderes übergeht, werden durch die rechten Seiten der Zielvariablen ersetzt. Dies betrifft die Regel $A \rightarrow V_a$, die ersetzt wird durch $A \rightarrow a$.“

S. 406, CNF; in Schritt 3 muss es genauer heißen: „Ersetze alle Produktionen der Form $X \rightarrow Y$, indem für jede Regel $Y \rightarrow w$ eine Regel $X \rightarrow w$ hinzugefügt wird. Die rechte Seite w kann hier nur noch aus einer oder zwei Variablen oder aus einem einzigen Terminalsymbol bestehen.“

S. 432, letzter Absatz. Es muss heißen: „Üblicherweise schreibt man hierfür $g(n) = \mathcal{O}(f(n)), \dots$ “ ($f(n)$ statt $f(N)$). (Danke an Enrico Winter für den Hinweis)

S. 433, Bsp. 11.8a, zweite Funktion: Es fehlt ein Plus-Zeichen, die beiden Zeilen müssen lauten:

$$f(n) = 2n^2 - 50n + 3 \rightarrow f(n) = \mathcal{O}(n^2) \text{ weil}$$

$$|2n^2 - 50n + 3| \leq 2n^2 + |50n| + 3 \leq 2n^2 + 50n^2 + 3n^2 = 55n^2 = |55n^2|, \text{ damit } c = 55, m = 1$$

(Danke an Gaetano Paganini für den Hinweis)

S. 447, im Abschnitt „EXSPACE und NEXSPACE“ muss es heißen: „Es wird vermutet, dass dies entsprechend auch für EXPTIME und NEXPTIME gilt, ...“. (Danke an Enrico Winter für den Hinweis)

S. 452, Gleichung (11.25): Hier sind die Parameter vertauscht. Es muss heißen:

$$x' = bx + a \quad .$$

S. 514, Beispiel 12.23: In der dritten Zeile der zu sortierenden Daten ist ein senkrechter Strich nach 06 zu viel. Die Zeile muss heißen:

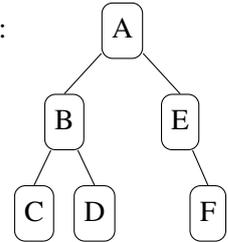
a: 44 55 | 12 42 | 18 94 | 06 67 Mischphase, $p = 2$

S. 517, Beispiel 12.25: In der zweiten Zeile der zu sortierenden Daten sind die beiden rechten senkrechten Striche um eine Stelle zu viel nach rechts gerückt. Die Zeile muss heißen:

b: 17 31 | 13 41 43 67 | 3 7 71 | 37 61

S. 527 Mitte. Die Inorder-Reihenfolge ist falsch. Es muss heißen: behandle den linken Teilbaum, behandle die Wurzel, behandle den rechten Teilbaum. (Danke an Tilo Strutz für den Hinweis)

S. 529, Bsp. 13.3: Beim gegebenen Baum sollte das Blatt F rechts sein, nicht links, also:



(Danke an Holger Hinrichs für den Hinweis)

S. 618, Zusammenfügen von Zeilen: Im Beispiel ist ein `\` zu viel; korrekt ist

```
Dies ist nur eine \
logische Zeile
```

S. 621, Tabelle 14.6: Die Wertebereiche von `unsigned short` und `unsigned int` (jeweils mit 16 Bit) sind falsch; außerdem die für Gleitkommazahlen. Korrekt ist:

Datentyp	Länge (Bit)	Bedeutung	Wertebereich
<code>unsigned short</code>	16	positive ganze Zahl	0 bis 65535
<code>unsigned int</code>	16	positive ganze Zahl	0 bis 65535
<code>float</code>	32	kurze Gleitpunktzahl	ca. $\pm 1,2 \cdot 10^{-38}$ bis $\pm 3,4 \cdot 10^{+38}$ (ca. 7 Stellen)
<code>double</code>	64	Gleitpunktzahl	ca. $\pm 2,2 \cdot 10^{-308}$ bis $\pm 1,8 \cdot 10^{+308}$ (ca. 15 Stellen)
<code>long double</code>	80	lange Gleitpunktzahl	ca. $\pm 3,4 \cdot 10^{-4932}$ bis $\pm 1,1 \cdot 10^{+4932}$ (ca. 18 Stellen)

S. 625, Bsp. 14.9: Die Definition von `complex` ist falsch; sie muss lauten:

```
typedef struct { float real; float imag; } complex;      // Typdefinition complex
complex z;
```

S. 625/626, „Ein erstes C-Programm“:

An Stelle von `a2+b2` muss es `a^2 + b^2` heißen.

Im nachfolgenden Text sind fehlerhafterweise Backslashes (`\`) vor `%f`.

S. 658, Tab. 14.15: Der Zeiger bei Index 8 (Krattler) muss 12 sein, nicht 4. (Danke an Holger Hinrichs für den Hinweis)

S. 9: Es fehlt der Abschnitt zur 3. Generation (Danke an Violetta V. Boehm für den Hinweis):

3. Generation – Zeitalter der integrierten Schaltkreise

Von den Transistoren ging man nun zu integrierten Schaltkreisen über. Mit deren Hilfe konnten bei erhöhter Leistungsfähigkeit noch kleinere und preiswertere Geräte entwickelt werden. Von der Firma Digital Equipment (DEC) wurden als typische Vertreter dieser Generation um 1960 die ersten Minicomputer (PDP 8) auf

den Markt gebracht, die auf einem Schreibtisch Platz finden konnten. IBM stellte 1964 den ersten Großrechner der Serie 360 vor. Diese unter der Leitung von Gene Amdahl entwickelte Computer-Familie stellte für lange Zeit die weltweit am meisten eingesetzten Rechner. Die Bezeichnung „360“ sollte symbolisieren, dass dieser Rechner „rundum“, also um 360 Grad, alle Ansprüche befriedigen könne. In dieser Zeit kamen auch zahlreiche weitere Programmiersprachen wie BASIC, PL/1, PASCAL etc. auf den Markt.