Algorithmen und algorithmisches Denken im Mathematikunterricht

FRANZ PAUER (UNIVERSITÄT INNSBRUCK)

In diesem Beitrag werden einige ausgewählte Algorithmen vorgestellt, die im Mathematikunterricht der Primarstufe und der Sekundarstufe erlernt werden: aus dem Unterricht der Primarstufe die Verfahren für das Rechnen mit Zahlen in Zifferndarstellung, aus dem der Sekundarstufe der Euklidische Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier positiver ganzer Zahlen, Gauß-Elimination zur Lösung von Systemen linearer Gleichungen und das Bisektionsverfahren zur näherungsweisen Bestimmung von Nullstellen stetiger Funktionen. Es ist wichtig, dass Algorithmen im Schulunterricht nicht nur eingeübt, sondern auch verständlich erklärt werden. Nur so kann den Schülerinnen und Schülern algorithmisches Denken vermittelt werden. Dazu ist es notwendig, die den Algorithmen zu Grunde liegenden Strategien und Ideen zu erläutern. Die österreichischen Lehrpläne sollten deutlicher darauf hinweisen.

1. Einleitung

Algorithmen und algorithmisches Denken sind derzeit in aller Munde. Diese Themen wecken auch Ängste. Darauf ist zum Beispiel Martin Kocher, seit 2021 österreichischer Minister für Arbeit und Wirtschaft, im Oktober 2020 im Vortrag „Angst vor Algorithmen? Verhaltensökonomische Aspekte der Digitalisierung“ am Institut für höhere Studien eingegangen (Kocher 2020). Allerdings gilt auch hier wie bei so vielen anderen Errungenschaften der Menschheit: vor Algorithmen braucht man keine Angst zu haben, aber vielleicht vor manchen Menschen, die sie verwenden.

Im Mathematikunterricht spielen Algorithmen eine große Rolle. Schon in der Volksschule werden jene Algorithmen besprochen, mit denen das Wort Algorithmus eng verbunden ist: das erste Buch über die Verfahren zum Rechnen mit natürlichen Zahlen in Zifferndarstellung wurde um das Jahr 825 n. Chr. von Muhamad al-Chwarizmi in Choresme im heutigen Usbekistan geschrieben. Aus *al-Chwarizmi* ist im Lauf der Zeit *Algorithmus* geworden.

Diese Algorithmen sind aber nicht die ältesten. Der Euklidische Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier natürlicher Zahlen (diesen braucht man zum optimalen Kürzen von Bruchzahlen) wird schon im 3. Jhd. v. Chr. in den Elementen des Euklid beschrieben. Dabei ist auch die zu Grunde liegende Strategie interessant: anstatt zu versuchen, ein Problem direkt zu lösen, ersetze es durch ein einfacheres, das aber dieselbe Lösung hat. Wiederhole das so lange, bis das Problem so einfach geworden ist, dass man die Lösung ohne weitere Überlegungen angeben kann.

Dieselbe Strategie wird auch beim Lösen von Systemen linearer Gleichungen und beim Bisektionsverfahren (zur näherungsweisen Bestimmung von Nullstellen stetiger Funktionen) erfolgreich angewandt. Dieses Verfahren liefert auch den Beweis des Zwischenwertsatzes, der besagt, dass die gesuchten Nullstellen unter den gegebenen Voraussetzungen immer existieren. Gibt es eine bessere Art, die Existenz einer Zahl mit gewissen Eigenschaften zu zeigen, als einen Algorithmus anzugeben, der sie berechnet?

Das Verfahren zum Lösen von Systemen linearer Gleichungen zeigt, dass die Einzelschritte von Algorithmen nicht immer genau vorgegeben sein müssen. Es können auch Wahlmöglichkeiten vorhanden sein, diese ermöglichen einen kreativen Umgang mit dem Verfahren.

Dieser Beitrag ist in die folgenden Abschnitte unterteilt:

- Was sind Algorithmen? Was bedeutet algorithmisches Denken?
- Eine Strategie für Algorithmen
- Der Euklidische Algorithmus - der älteste Algorithmus
- Gauß-Elimination - ein Algorithmus mit Wahlmöglichkeiten

Ich danke Manfred Borovcnik für das sorgfältige Lesen des Manuskripts und für viele Beiträge zur Verbesserung dieses Textes.

- Algorithmen zum Rechnen mit natürlichen Zahlen in Zifferndarstellung - die ersten Algorithmen in der Schule
- Das Bisektionsverfahren - ein Algorithmus zur Berechnung von Nullstellen stetiger Funktionen
- Schlussbemerkungen

2. Was sind Algorithmen? Was bedeutet algorithmisches Denken?

Algorithmen sind Verfahren, mit denen eine *Klasse von Problemen* gelöst werden kann.

Algorithmen müssen endlich und korrekt sein, das heißt:

- aus den Eingabedaten erhält man in endlich vielen Schritten Ausgabedaten und
- diese sind eine Lösung des betrachteten Problems.

Zur Lösung einer Klasse von Problemen kann es mehrere Algorithmen - mit verschiedener Effizienz - geben. Es liegt nahe, jenen mit geringstem Aufwand zu wählen.

Algorithmisches Denken bedeutet: Algorithmen zu verstehen, erklären, beurteilen, modifizieren und entwickeln.

Algorithmen und algorithmisches Denken sind im Lehrplan der AHS in mehreren Unterrichtsfächern verankert, zum Beispiel findet man

- beim Unterrichtsfach Digitale Grundbildung *Algorithmen nachvollziehen und entwerfen*
- beim Unterrichtsfach Informatik *Algorithmen erklären, entwerfen, implementieren, testen* und
- beim Unterrichtsfach Darstellende Geometrie *algorithmische Denkfähigkeit entwickeln und vertiefen*.

Im Mathematikunterricht lernt man viele Algorithmen kennen, manche bereits in der Volksschule. Sie sollten so unterrichtet werden, dass damit algorithmisches Denken gefördert wird. Algorithmen sollten also nicht nur „eingeübt“ (oder gar „eingedrillt“) werden, sondern auch verständlich erklärt (zugrunde liegende Strategien und Ideen angeben, Korrektheit begründen) werden! Im Lehrplan der AHS wird das im Abschnitt für das Unterrichtsfach Mathematik eher verschwiegen. Das Wort Algorithmus kommt nur bei der Erklärung des Begriffs *formal-operatives Arbeiten* vor, dort werden *Aktivitäten, die auf Algorithmen beruhen* erwähnt.

3. Eine Strategie für Algorithmen

Mathematische Probleme kann man sehr grob in zwei Klassen einteilen:
Einfache Probleme, das sind Probleme, die man durch „Hinschauen“ (ohne zu rechnen) lösen kann.
Nicht-einfache Probleme, das sind die anderen Probleme.

Beispiele für „einfache Probleme“ sind

Berechne ggT(123456, 123456)!
Die Lösung ist 123456.

Bestimme die Lösungsmenge der Gleichung $3x+4y=7$!
Die Lösungsmenge ist $\{(1,1)+t\cdot(4,-3)\,|\,t\in\mathbb{R}\}$.

Bestimme die Lösungsmenge des Systems linearer Gleichungen
$1\cdot x+0\cdot y=3$
$0\cdot x+1\cdot y=-8$!
Die Lösungsmenge ist $\{(3,-8)\}$.

Eine wichtige Strategie, ein Problem zu lösen ist: ersetze das Problem durch ein anderes, das einfacher ist, aber dieselbe Lösung hat. Wiederhole das zielgerichtet solange, bis ein einfaches Problem erreicht wird.

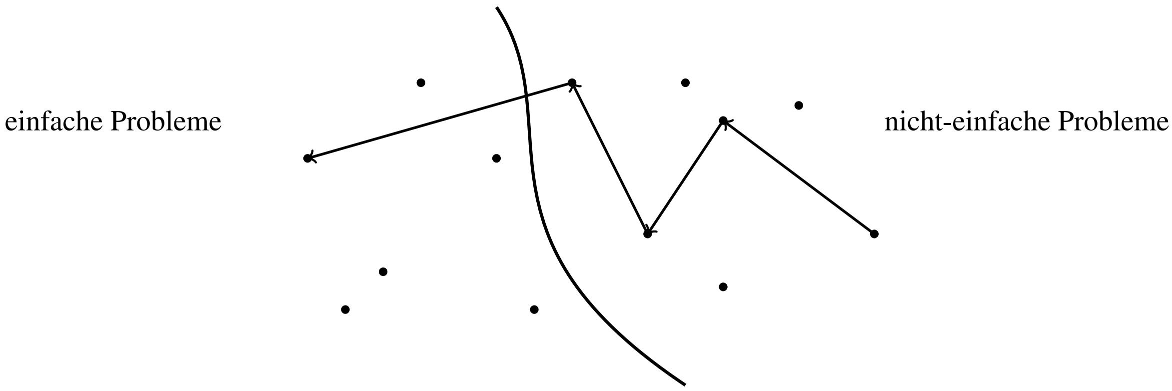

Abbildung 1: Übergang von einem nicht-einfachen zu einem einfachen Problem

4. Der Euklidische Algorithmus - der älteste Algorithmus

Der Euklidische Algorithmus ist der älteste Algorithmus der Mathematik. Er wird in den Elementen des Euklid um 300 v. Chr. beschrieben, war aber schon früher bekannt.

Man löst damit die folgende Aufgabe:
Gegeben sind positive ganze Zahlen a und b. Bestimme die größte ganze Zahl, die beide teilt. Diese Zahl heißt *größter gemeinsamer Teiler von a und b*, man schreibt dafür kurz $\mathrm{ggT}(a,b)$.

Was macht man mit dem größten gemeinsamen Teiler zweier natürlicher Zahlen? Warum möchte man ihn berechnen? Seine wichtigste Anwendung ist das optimale Kürzen von Bruchzahlen. Um Bruchzahlen durch möglichst kleine Zähler und Nenner darzustellen, sollte man immer - insbesondere nach jeder Rechenoperation - durch den größten gemeinsamen Teiler von Zähler und Nenner kürzen. Es ist daher sinnvoll, die Berechnung des ggT erst nach der Einführung der Bruchzahlen und der Besprechung des Kürzens zu unterrichten.

Die Strategie des Euklidischen Algorithmus ist die in Abschnitt 3 besprochene.
Wann ist der ggT zweier Zahlen besonders leicht zu bestimmen? Wenn die zwei Zahlen gleich sind: $\mathrm{ggT}(a,a) = a$.

Die folgende Eigenschaft des ggT ermöglicht es, jede Berechnung eines ggT auf diesen einfachen Fall zurückzuführen:
Für positive ganze Zahlen a und b mit $a \geq b$ ist

$$\mathrm{ggT}(a,b) = \mathrm{ggT}(a-b,b)\,.$$

Denn: Ist t ein Teiler von a und von b, dann gibt es positive ganze Zahlen u und v so, dass $t \cdot u = a$ und $t \cdot v = b$ ist. Wegen $t \cdot u \pm t \cdot v = t \cdot (u \pm v)$ ist jeder Teiler von a und b auch ein Teiler von $a+b$ und $a-b$. Ein gemeinsamer Teiler von a und b ist daher auch ein gemeinsamer Teiler von $a-b$ und b. Umgekehrt ist jeder gemeinsame Teiler von $a-b$ und b auch ein gemeinsamer Teiler von $a = (a-b)+b$ und b. Daraus folgt die Behauptung.

Zum Beispiel ist

$$\mathrm{ggT}(713,589) = \mathrm{ggT}(713-589,589) = \mathrm{ggT}(124,589)\,.$$

Im Euklidischen Algorithmus wird solange die größere Zahl durch die Differenz der größeren und der kleineren ersetzt, bis die zwei Zahlen gleich sind.

Also: gehe von (a,b) zu

$$(\max(a,b)-\min(a,b),\min(a,b))$$

über, bis die zwei Zahlen gleich sind.

Jedesmal wird die größere der zwei positiven ganzen Zahlen kleiner, also müssen die zwei Zahlen nach endlich vielen Wiederholungen dieses Schrittes gleich sein.

Dieser Algorithmus kann sehr einfach in einer Programmiersprache dargestellt werden, zum Beispiel:

$$\begin{gathered}\text{while } a \neq b \text{ do}\\ c := \max(a,b),\ d := \min(a,b),\ a := c-d,\ b := d;\end{gathered}$$

Beipiel:
Wir berechnen mit diesem Verfahren rasch (durch einige Subtraktionen) den größten gemeinsamen Teiler der Zahlen 713 und 598.

$$\mathrm{ggT}(713,589) = \mathrm{ggT}(589,124) = \mathrm{ggT}(465,124) =$$

$$= \mathrm{ggT}(341,124) = \mathrm{ggT}(217,124) = \mathrm{ggT}(124,93) =$$

$$= \mathrm{ggT}(93,31) = \mathrm{ggT}(62,31) = \mathrm{ggT}(31,31) = 31$$

Wer in der Lage ist, dreistellige Zahlen im Kopf zu subtrahieren, kann diese Berechnung auch im Kopf ausführen, weil man sich immer nur das zuletzt berechnete Zahlenpaar merken muss.

Wenn a viel größer als b ist, wird im Algorithmus b mehrfach von a abgezogen. Das kann man zu einer Division mit Rest von a durch b zusammenfassen und a gleich durch seinen Rest nach Division durch b ersetzen. Zuvor muss noch vereinbart werden, dass $\mathrm{ggT}(a,0) = a$ ist.
Dann hat der Euklidische Algorithmus die folgende Form: ersetze solange die größere Zahl durch ihren Rest nach Division mit Rest durch die kleinere, bis eine der zwei Zahlen 0 ist.

Im Beispiel verkürzt sich dann die Berechnung von $\mathrm{ggT}(713,589)$ zu

$$\mathrm{ggT}(713,589) = ggT(589,124) = \mathrm{ggT}(124,93) = \mathrm{ggT}(93,31) = \mathrm{ggT}(31,0) = 31.$$

Der Euklidische Algorithmus ist nicht nur das älteste, sondern auch das effizienteste Verfahren, den ggT zweier positiver ganzer Zahlen zu berechnen. Er ist in jedem Computeralgebrasystem implementiert und wichtig für das effiziente Rechnen mit Bruchzahlen. Man kann damit auch schnell das kleinste gemeinsame Vielfache zweier positiver ganzer Zahlen a und b berechnen:

$$\mathrm{kgV}(a,b) = \frac{a \cdot b}{\mathrm{ggT}(a,b)}\,.$$

Den ggT von drei oder mehr Zahlen berechnet man sukzessive:

$$ggT(a,b,c) = ggT(a, ggT(b,c))\,,$$

man wendet also zweimal den Euklidischen Algorithmus an.

Die Berechnung des ggT ist im Lehrplan für die 6. Schulstufe vorgeschrieben. Es ist erfreulich, dass die Lehrbücher (Humenberger 2017) und (Salzger et al. 2022) dafür den Euklidischen Algorithmus verwenden. Viele andere Lehrbücher berechnen den ggT als Produkt der gemeinsamen Primfaktoren. Dieses Verfahren ist sehr ineffizient, weil es die Zerlegung von ganzen Zahlen in ihre Primfaktoren erfordert. Diese ist so aufwändig, dass zum Beispiel das RSA-Verfahren zur Verschlüsselung mit öffentlichem Schlüssel (siehe zum Beispiel (Walz 2023) oder (Pauer 2019)) darauf beruht, dass die Faktorisierung einer großen Zahl in Primfaktoren in vernünftiger Zeit nicht möglich ist. Den ggT zweier Zahlen wird man nur dann als Produkt ihrer gemeinsamen Primfaktoren berechnen, wenn die zwei Zahlen schon als Produkte von Primzahlen dargestellt sind. Im Beispiel oben also $\mathrm{ggT}(23 \cdot 31, 19 \cdot 31) = 31$. Allerdings sind Primzahlen gar nicht Thema der Lehrpläne der Sekundarstufe 1. Ausführlichere Information dazu findet man zum Beispiel in (Pauer und Stampfer 2013).

5. Gauß-Elimination - ein Algorithmus mit Wahlmöglichkeiten

Ein anderer Algorithmus, der die Strategie von Abschnitt 3 verwendet, ist das Verfahren zur Lösung von Systemen linearer Gleichungen durch *Gauß-Elimination*. Dieser Algorithmus stammt nicht von Gauß, sondern wurde erstmals in China (ca. 150 v. Chr.) publiziert und in Europa von Rolle und Newton (ca. 1690 n. Chr.) wiederentdeckt, siehe (Grcar 2011).

Wir beschränken uns hier auf Systeme von 2 linearen Gleichungen mit 2 Unbekannten, also auf die folgenden Aufgaben:
Gegeben sind Zahlen a,b,c,d,e,f mit $(a,b,d,e) \neq (0,0,0,0)$. Gesucht sind alle Zahlenpaare (x,y) mit

$$ax + by = c$$

$$dx + ey = f.$$

Um sie zu lösen, ersetzen wir das System linearer Gleichungen durch ein einfacheres mit derselben Lösungsmenge und zwar so oft, bis wir eines erhalten, dessen Lösungsmenge ohne weitere Rechnung angeschrieben werden kann.

Welche Gleichungssysteme haben diese Eigenschaft, sind also *einfach* ? Wir unterscheiden drei Typen:

- Die Lösungsmenge von
$$\begin{array}{ccc} x & & = & c \\ & y & = & f \end{array}$$
ist $\{(c,f)\}$.

- Die Lösungsmenge von
$$\begin{array}{ccccc} ax & + & by & = & c \\ & & 0 & = & 0 \end{array}$$
ist $\{(\frac{a \cdot c}{a^2+b^2}, \frac{b \cdot c}{a^2+b^2}) + t \cdot (-b,a) \,|\, t \in \mathbb{R}\}$.

- Die Lösungsmenge von
$$\begin{array}{ccccc} ax & + & by & = & c \\ & & 0 & = & 1 \end{array}$$
ist leer.

Wir verwenden eine nützliche Eigenschaft von Systemen linearer Gleichungen:
Durch die folgenden drei *elementaren Umformungen* wird ein System linearer Gleichungen in ein anderes mit gleicher Lösungsmenge übergeführt.

Multipliziere auf beiden Seiten einer Gleichung mit einer Zahl $\neq 0$.

Vertausche die zwei Gleichungen.

Ersetze eine Gleichung durch die Summe der zwei Gleichungen.

Man kann leicht zeigen, dass man mit diesen elementaren Umformungen jedes System in eines von einem der drei oben angeführten Typen überführen kann. Die Reihenfolge dieser Umformungen wählt man so, dass man einem dieser Typen mit jeder Umformung etwas näher kommt (siehe zum Beispiel Pauer 2018).

Wir betrachten das folgende Beispiel:

$$2x+3y=4$$
$$-x+5y=2$$

Für die Wahl der Umformungen gibt es mehrere Möglichkeiten, etwa:

- Multipliziere die zweite Gleichung mit 2

$$\begin{array}{rcrcr} 2x & + & 3y & = & 4 \\ -2x & + & 10y & = & 4 \end{array}$$

- Ersetze die erste Gleichung durch die Summe der ersten und der zweiten Gleichung

$$\begin{array}{rcrcr} & & 13y & = & 8 \\ -2x & + & 10y & = & 4 \end{array}$$

- Multipliziere die erste Gleichung mit $\frac{1}{13}$

$$\begin{array}{rcrcr} & & y & = & \frac{8}{13} \\ -2x & + & 10y & = & 4 \end{array}$$

- Multipliziere die zweite Gleichung mit $-\frac{1}{10}$

$$\begin{array}{rcrcr} & & y & = & \frac{8}{13} \\ \frac{1}{5}x & - & y & = & -\frac{2}{5} \end{array}$$

- Ersetze die zweite Gleichung durch die Summe der ersten und der zweiten

$$\begin{array}{rcrcr} & & y & = & \frac{8}{13} \\ \frac{1}{5}x & & & = & \frac{14}{65} \end{array}$$

- Multipliziere die zweite Gleichung mit 5 und vertausche die zwei Gleichungen

$$\begin{array}{rcrcr} x & & & = & \frac{14}{13} \\ & & y & = & \frac{8}{13} \end{array}$$

Dieses System hat die Lösungsmenge $\{(\frac{14}{13}, \frac{8}{13})\}$, daher ist das auch die Lösungsmenge des ursprünglichen Systems (und aller durch die Umformungen erhaltenen Systeme).

Auch für das System

$$2x+3y=4$$
$$-4x-6y=-8$$

gibt es mehrere Möglichkeiten, es mittels Gauß-Elimination zu lösen, etwa:

- Multipliziere die erste Gleichung mit 2

$$4x+6y=8$$
$$-4x-6y=-8$$

- Ersetze die zweite Gleichung durch die Summe der ersten und der zweiten Gleichung

$$\begin{array}{rcrcl} 4x & + & 6y & = & 8 \\ & & 0 & = & 0 \end{array}$$

Die Lösungsmenge dieses Systems ist $\{(2,0)+t\cdot(-6,4)\,|\,t\in\mathbb{R}\}$.

Bei diesem Algorithmus sind die Einzelschritte nicht vorgegeben, sie können je nach Situation und Vorliebe von Schülerinnen und Schülern gewählt werden. Das sollte man auch zulassen, um so das selbständige Denken zu fördern. Es ist nicht sinnvoll, im Unterricht eine bestimmte Abfolge der elementaren Umformungen vorzugeben und je nach vorgegebener Abfolge das Verfahren noch eigens zu benennen, zum Beispiel *Einsetzungsverfahren*, *Gleichsetzungsverfahren*, $\dots$. Alle diese Verfahren sind nur Varianten der Gauß-Elimination.

6. Algorithmen zum Rechnen mit natürlichen Zahlen in Zifferndarstellung - die ersten Algorithmen in der Schule

In diesem Abschnitt werden die Algorithmen für die vier Grundrechnungsarten (für natürliche Zahlen in Zifferndarstellung) besprochen. Diese vier Algorithmen haben eine gemeinsame Grundstruktur, weisen aber auch einige Unterschiede auf. Zunächst wird die Problemstellung präzisiert, dann wird der Reihe nach auf die Verfahren für Addition, Subtraktion, Multiplikation und Division mit Rest eingegangen.

6.1. Rechnen mit natürlichen Zahlen in Zifferndarstellung

Was ist mit

$$\textit{Berechne}\ \ 378+253,\ 987-234\ \ \textit{und}\ \ 345\cdot 67\ \ \textit{!}$$

gemeint? Zunächst ist nicht klar, was zu berechnen ist. Die Zahlen $378+253$, $987-234$ und $345\cdot 67$ sind ja eindeutig bestimmt und als Summe, Differenz, Produkt von zwei anderen dargestellt.

Die Aufforderung oben ist eine Kurzform von

$$\textit{Berechne die Zifferndarstellung (zur Basis } 10\textit{) von}\ \ 378+253,\ 987-234\ \textit{und}\ \ 345\cdot 67\ \ \textit{!}$$

Grundlage dafür ist der folgende Satz, der für jede natürliche Zahl $b>1$ gilt.

Zu jeder positiven ganzen Zahl a gibt es eindeutig bestimmte natürliche Zahlen $n, z_n, z_{n-1}, \dots, z_1, z_0$ *so, dass*

$$0 \le z_n, z_{n-1}, \dots, z_1, z_0 < b,\ z_n \neq 0$$

und

$$a = z_n b^n + z_{n-1} b^{n-1} + \dots + z_1 b^1 + z_0$$

ist.

Diese Darstellung von a heißt ihre *Zifferndarstellung zur Basis b*, die Zahlen $z_n, z_{n-1}, \dots, z_1, z_0$ heißen *Ziffern* von a (bezüglich der Basis b).

Wenn b fest gewählt ist (im Folgenden wird b immer 10 sein) schreibt man dafür kurz

$$a = z_n z_{n-1} \dots z_1 z_0\,.$$

Zum Beispiel wird die Zahl $3\cdot 10^2 + 7\cdot 10^1 + 8$ kurz als 378 angeschrieben.

Zur formalen Beschreibung der Algorithmen für die Addition und die Subtraktion ist es von Vorteil, auf die Bedingung $z_n \neq 0$ zu verzichten. Dann kann man annehmen, dass zwei betrachtete Zahlen gleich viele Ziffern haben, indem man bei einer davon einige Nullen als führende Ziffern hinzunimmt. Zum Beispiel hat dann 0012 gleich viele Ziffern wie 3456.

Die Zifferndarstellung einer Zahl ist eine *Zusatzinformation* über diese und erweist sich als sehr nützlich, zum Beispiel für das effiziente Rechnen mit natürlichen Zahlen oder für das Entscheiden, welche von zwei Zahlen die größere ist.

Man kann mit Zahlen aber auch dann rechnen, wenn sie anders dargestellt sind, zum Beispiel durch römische Zahlzeichen. Man bedenke etwa, dass der Mathematiker und Ingenieur Archimedes, der im 3. Jhd. v. Chr. lebte, für seine Berechnungen ohne die heutige Zifferndarstellung auskommen musste.

Um das Jahr 825 n. Chr. hat Muhammad al-Chwarizmi ein Buch über das Rechnen mit „indischen Ziffern“ veröffentlicht. Darin werden Algorithmen für das Rechnen mit Zahlen in Zifferndarstellung vorgestellt. Diese Algorithmen lernt man heute in der Volksschule. Sie sind zwar nicht die ältesten, haben aber zum Wort „Algorithmus“ geführt, siehe zum Beispiel (Ziegenbalg et al. 2010).

Alle diese Rechenverfahren sind für jede Basis $b > 1$ gültig, insbesondere auch für die Basis 2, bezüglich der natürliche Zahlen in den meisten Computern dargestellt werden. Im Folgenden beschränken wir uns aber auf die Basis 10.

Die Algorithmen für die Grundrechnungsarten können grob so beschrieben werden:

- Gegeben sind zwei natürliche Zahlen in Zifferndarstellung.
  Gesucht ist die Zifferndarstellung ihrer Summe, ihrer Differenz (falls diese nicht negativ ist), ihres Produktes oder ihres ganzzahligen Quotienten sowie des Restes.
- Es werden einfache „Grundschritte“ mehrfach ausgeführt, um die Aufgabe schrittweise zu lösen.
- Dazu werden die Zifferndarstellung und die Regeln für das Rechnen mit natürlichen Zahlen verwendet.
- Die Grundschritte können nicht mit Hilfe des Algorithmus ausgeführt werden, sie müssen vorab erlernt werden.

Beim Unterrichten dieser Algorithmen ist es wichtig, die Grundschritte jeder Rechenart zuerst gut einzuüben und dann zu erklären, warum man durch deren mehrfaches Anwenden das richtige Ergebnis erhält.

6.2. Additionsalgorithmus

- Gegeben sind zwei natürliche Zahlen x und y in Zifferndarstellung. Gesucht ist die Zifferndarstellung ihrer Summe $x+y$.
- Die Grundschritte sind die Rechenoperationen des „kleinen Eins plus Eins“, also die hundert Additionen $x+y$ für $0 \leq x, y < 10$.
  Die Summe von zwei einstelligen Zahlen ist höchstens zweistellig und kleiner oder gleich 18. Ihre Zehnerziffer heißt „Übertrag“ (dieser ist 1, wenn die Summe größer als 9 ist, sonst 0).
- Die Ziffern mit gleichem Index der zwei Zahlen werden - beginnend mit den nullten Ziffern - addiert, ab den Ziffern mit Index 1 wird auch noch der Übertrag der vorangegangenen Addition addiert.
- Der Additionsalgorithmus verwendet die folgenden Rechenregeln: Summanden können vertauscht werden, bei mehrfacher Addition können Klammern weggelassen werden, man kann Herausheben und Ausmultiplizieren.

Wir betrachten zuerst ein Beispiel:

Addiere 378 und 253 !

$$378+253 = (3\cdot 10^2+7\cdot 10^1+8)+(2\cdot 10^2+5\cdot 10^1+3) =$$
$$= (3+2)\cdot 10^2+(7+5)\cdot 10^1+(8+3) =$$

$$= 5 \cdot 10^2 + (10+2) \cdot 10^1 + (10^1 + 1) =$$
$$= 5 \cdot 10^2 + 1 \cdot 10^2 + 2 \cdot 10^1 + 1 \cdot 10^1 + 1 =$$
$$= 6 \cdot 10^2 + 3 \cdot 10^1 + 1 = 631$$

Der Additionsalgorithmus kann formal so formuliert werden:

> Gegeben: Zahlen x mit Ziffern $x_n, x_{n-1}, \ldots, x_1, x_0$ und y mit Ziffern $y_n, y_{n-1}, \ldots, y_1, y_0$.
>
> Gesucht: die Ziffern $z_{n+1}, z_n, z_{n-1}, \ldots, z_1, z_0$ von $x+y$.
>
> Berechne zuerst u_0 und z_0 mit $x_0 + y_0 = u_0 \cdot 10 + z_0$ und dann der Reihe nach
> für $i = 1, \ldots, n$ die Zahlen z_i und u_i mit $x_i + y_i + u_{i-1} = u_i \cdot 10 + z_i$.
> Setze $z_{n+1} := u_n$.

Technisch kann dieser Algorithmus durch ein *Addierwerk* (zumeist aber mit Ziffern bezüglich der Basis 2) realisiert werden (Abbildung 2):

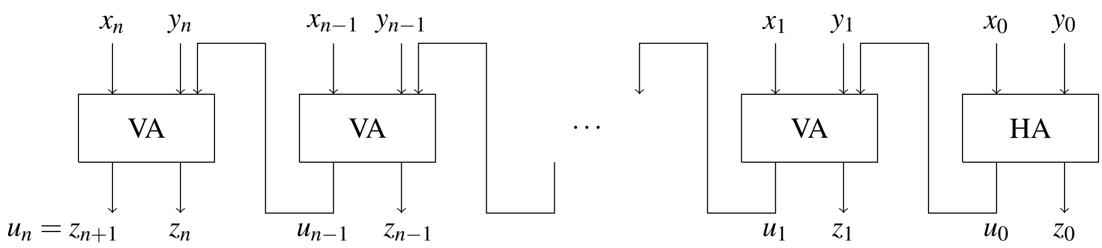

Abbildung 2: Skizze eines Addierwerkes

Dabei steht HA bzw. VA für die elektronischen Bauteile Halbaddierer bzw. Volladdierer. Der Halbaddierer gibt nach Eingabe zweier Ziffern die Ziffern ihrer Summe aus, der Volladdierer gibt nach Eingabe dreier Ziffern die Ziffern ihrer Summe aus.

6.3. Subtraktionsalgorithmus

- Gegeben sind zwei natürliche Zahlen x und y in Zifferndarstellung mit $x \geq y$. Gesucht ist die Zifferndarstellung ihrer Differenz $x - y$.
- Die Grundschritte sind die insgesamt 100 Subtraktionen $x - y$ für $0 \leq y \leq x < 10$ und $(10+x) - y$ für $0 \leq x < y < 10$.
  Das Ergebnis ist immer einstellig, der „Übertrag" ist im ersten Fall 0, im zweiten Fall 1.
- Diese Grundschritte werden zuerst für x_0 und y_0 und dann der Reihe nach für x_i und $y_i + u_{i-1}$ (oder für $x_i - u_{i-1}$ und y_i) ausgeführt ($i > 0$). Dabei ist u_{i-1} der Übertrag aus dem vorangegangenen $(i-1)$-ten Schritt.
- Der Subtraktionsalgorithmus verwendet die folgenden Rechenregeln: Summanden können vertauscht werden, bei mehrfacher Addition können Klammern weggelassen werden, man kann Herausheben und Ausmultiplizieren.

Wir betrachten zuerst ein Beispiel:

Subtrahiere 378 von 524 !

$$524 - 378 = (5 \cdot 10^2 + 2 \cdot 10^1 + 4) - (3 \cdot 10^2 + 7 \cdot 10^1 + 8) =$$
$$= (5-3) \cdot 10^2 + (2-7) \cdot 10^1 + (4-8) =$$
$$= (5-3-1) \cdot 10^2 + (10+2-7-1) \cdot 10^1 + (10^1 + 4 - 8) =$$
$$= 1 \cdot 10^2 + 4 \cdot 10^1 + 6 = 146\,.$$

Der Subtraktionsalgorithmus kann formal so formuliert werden:

Gegeben: Zahlen x mit Ziffern $x_n, x_{n-1}, \dots, x_1, x_0$ und $y \leq x$ mit Ziffern $y_n, y_{n-1}, \dots, y_1, y_0$.

Gesucht: die Ziffern $z_n, z_{n-1}, \dots, z_1, z_0$ von $x - y$.

Falls $x_0 \geq y_0$ ist, berechne $z_0 := x_0 - y_0$ und setze $u_0 := 0$.
Falls $x_0 < y_0$ ist, berechne $z_0 := 10 + x_0 - y_0$ und setze $u_0 := 1$.
Für $i = 1, \dots, n$:
Falls $x_i - u_{i-1} \geq y_i$ ist, berechne $z_i := x_i - u_{i-1} - y_i$ und setze $u_i := 0$.
Falls $x_i - u_{i-1} < y_i$ ist, berechne $z_i := 10 + x_i - u_{i-1} - y_i$ und setze $u_i := 1$.

6.4. Multiplikationsalgorithmus

Die Multiplikation mit einer natürlichen Zahl y mit einer natürlichen Zahl x ist als mehrfache Addition definiert.
Ein erster Algorithmus zum Multiplizieren ist daher: Addiere $(y-1)$-mal x zu x.

Mithilfe der Zifferndarstellung erhält man einen effizienteren Algorithmus:

- Gegeben sind zwei natürliche Zahlen x und y in Zifferndarstellung. Gesucht ist die Zifferndarstellung ihres Produktes $x \cdot y$.
- Die Grundschritte sind die die Multiplikationen mit Zehnerpotenzen $(10, 10^2, 10^3, \ ,\dots\ )$ und die Multiplikationen des „kleinen Einmaleins", also $x \cdot y$ für $0 \leq x, y < 10$. Diese Produkte berechnet man mit dem ersten Algorithmus oder man prägt sich - zum schnelleren Rechnen - das kleine Einmaleins ein.
  Das Produkt von einstelligen Zahlen ist höchstens zweistellig, der „Übertrag" ist die Zehnerziffer dieses Produktes.
- Für alle j: Multipliziere x mit y_j durch mehrfaches Verwenden des kleinen Einmaleins und der Multiplikation mit Zehnerpotenzen (für alle i berechne $x_i \cdot y_j \cdot 10^i$ und addiere alle diese Produkte) und multipliziere $x \cdot y_j$ mit 10^j.
  Addiere für alle j die Produkte $x \cdot y_j \cdot 10^j$.
- Der Multiplikationsalgorithmus verwendet die folgenden Rechenregeln: Summanden können vertauscht werden, Faktoren können vertauscht werden, bei mehrfacher Addition oder mehrfacher Multiplikation können Klammern weggelassen werden, man kann Herausheben und Ausmultiplizieren.
  Weiters wird der Additionsalgorithmus verwendet.

Beispiel:
Berechne die Zifferndarstellung von $345 \cdot 67$!

$$345 \cdot 67 = 345 \cdot (6 \cdot 10 + 7) = (345 \cdot 6) \cdot 10 + 345 \cdot 7 =$$

$$= (3 \cdot 10^2 + 4 \cdot 10 + 5) \cdot 6 \cdot 10 + (3 \cdot 10^2 + 4 \cdot 10 + 5) \cdot 7 =$$

$$= (3 \cdot 6 \cdot 10^2 + 4 \cdot 6 \cdot 10 + 5 \cdot 6) \cdot 10 + (3 \cdot 7 \cdot 10^2 + 4 \cdot 7 \cdot 10 + 5 \cdot 7) =$$

$$= ((10+8) \cdot 10^2 + (2 \cdot 10 + 4) \cdot 10 + 3 \cdot 10) \cdot 10 + ((2 \cdot 10 + 1) \cdot 10^2 + (2 \cdot 10 + 8) \cdot 10 + (3 \cdot 10 + 5)) =$$

$$= (10^3 + (8+2) \cdot 10^2 + (4+3) \cdot 10) \cdot 10 + (2 \cdot 10^3 + (1+2) \cdot 10^2 + (8+3) \cdot 10 + 5) =$$

$$= (10^3 + 10 \cdot 10^2 + 7 \cdot 10) \cdot 10 + (2 \cdot 10^3 + 3 \cdot 10^2 + (10+1) \cdot 10 + 5) =$$

$$= (2 \cdot 10^3 + 7 \cdot 10) \cdot 10 + (2 \cdot 10^3 + 4 \cdot 10^2 + 1 \cdot 10 + 5) =$$

$$= 20700 + 2415 = 23115$$

Man kann das übersichtlich mit den folgenden Schreibweisen darstellen:

$$
\begin{array}{ccccc}
 & 3 & 4 & 5 & \cdot\ 6\ 7 \\
\hline
2 & 0 & 7 & 0 & \\
 & 2 & 4 & 1 & 5 \\
\hline
2 & 3 & 1 & 1 & 5
\end{array}
\qquad \text{oder} \qquad
\begin{array}{ccccc}
 & 3 & 4 & 5 & \cdot\ 6\ 7 \\
\hline
 & 2 & 4 & 1 & 5 \\
2 & 0 & 7 & 0 & \\
\hline
2 & 3 & 1 & 1 & 5
\end{array}
$$

Die Schreibweise links ist in vielen europäischen Ländern üblich, die rechts zum Beispiel in den USA oder in Syrien. Das übliche Weglassen der letzten Ziffer 0 von $345 \cdot 60$ (links in der zweiten Zeile bzw. rechts in der dritten Zeile) bringt wenig Schreibersparnis, aber erschwert vielleicht das Verständnis des Verfahrens. Schreibt man die 0 an, sieht man sofort, dass sich die zwei Schreibweisen nur durch die Reihenfolge der Summanden der abschließenden Addition unterscheiden.

Daraus ergibt sich der sachgerechte Aufbau des Unterrichts der Multiplikation:

- Zuerst werden die Grundschritte kleines Einmaleins und Multiplikation mit $10, 10^2, 10^3, \ldots$ eingeübt.
- Dann die Multiplikation einer Zahl mit einer einstelligen Zahl
- und schließlich der allgemeine Fall.

6.5. Algorithmus für die Division mit Rest

Die Division mit Rest einer natürlichen Zahl x durch eine positive ganze Zahl y berechnet natürliche Zahlen q und r so, dass $x = q \cdot y + r$ und $r < y$ ist. Die Zahlen q und r sind eindeutig bestimmt und heißen *ganzzahliger Quotient* und *Rest* von x nach Division mit Rest durch y.

Ein erster Algorithmus zum Dividieren mit Rest ist: subtrahiere y so oft wie möglich von x (solange die Differenz nicht negativ wird). Der ganzzahlige Quotient ist dann die Anzahl der Subtraktionen und der Rest ist die letzte (nicht negative) Differenz.
Division mit Rest von natürlichen Zahlen ist also mehrfache Subtraktion.

Beispiel:
Dividiere 53 mit Rest durch 24!

$$53 - 24 = 29, \quad 29 - 24 = 5 \text{ und } 5 < 24\,,$$

also ist

$$53 = 2 \cdot 24 + 5\,,$$

somit ist der ganzzahlige Quotient 2 und der Rest gleich 5.

Anmerkung: Statt $x = q \cdot y + r$ und $r < y$ schreibt man oft $x : y = q$, Rest r. Man nennt dann x Dividend und y Divisor. Division mit Rest ist aber eine Rechenoperation für natürliche (und ganze) Zahlen und darf nicht mit einer Division (der Umkehrung der Multiplikation) verwechselt werden. Diese ist eine Rechenoperation für rationale, reelle und komplexe Zahlen. Bei einer Division mit Rest werden zwei Zahlen berechnet (ganzzahliger Quotient und Rest), bei einer Division jedoch nur eine. Bei einer Division mit Rest spielt die Ordnungsrelation $<$ eine wichtige Rolle, bei einer Division aber keine.

Für die Entwicklung eines effizienten Algorithmus muss man sich überlegen, wie mit Hilfe der Zifferndarstellung Subtraktionen eingespart werden können. Die Idee dazu ist einfach:

Wenn man 24 zwei-mal von 53 abziehen kann, dann mindestens zwanzig-mal von 530 und mindestens zweihundert-mal von 5300.

Denn, wenn

$$53 = 2 \cdot 24 + 5$$

ist, dann ist

$$530 = 20 \cdot 24 + 50$$

und

$$5300 = 200 \cdot 24 + 500\,.$$

Das legt den folgenden Algorithmus für die „schriftliche Division (mit Rest)“ nahe, den man in der Volksschule kennen lernt.

- Gegeben: natürliche Zahlen $x = x_n x_{n-1} \dots x_1 x_0$ und $y \neq 0$ in Zifferndarstellung.
  Gesucht: die Zifferndarstellung der natürlichen Zahlen q und r mit

$$x = q \cdot y + r \text{ und } r < y.$$

- Die Grundschritte sind die Divisionen mit Rest mit *einstelligem Quotienten* (das heißt, der Dividend ist kleiner als 10-mal der Divisor). Beispiele für Grundschritte sind 4321 : 567 oder 61 : 7, aber nicht 53 : 4 (weil $53 > 10 \cdot 4$ ist).
  Die Grundschritte werden mit dem ersten Algorithmus (mehrfaches - höchstens 9-maliges - Subtrahieren) ausgeführt (oder durch „Versuch und Irrtum“, das heißt: Abschätzen der nötigen Anzahl der Subtraktionen, Überprüfen durch Multiplikation und eventuell Korrektur der Abschätzung).
  Im Unterschied zu den ersten drei Rechenoperationen kann man sich bei der Division mit Rest die (unendlich vielen) Grundschritte nicht alle einprägen.

- Das effiziente Verfahren:
  - Setze $q := 0$.
  - ∗ Es sei k die durch $x = s \cdot 10^k + t,\ \ y \leq s < 10 \cdot y$ und $t < 10^k$ eindeutig bestimmte Zahl. Dann ist $t = x_{k-1} x_{k-2} \dots x_1 x_0$ und $s = x_n x_{n-1} \dots x_{k+1} x_k$.
    Dividiere nun s (durch mehrfache, höchstens 9-fache Subtraktion) mit Rest durch y und erhalte $s = q_k \cdot y + r_k$ und $r_k < y$.
    (Es ist $x = q_k \cdot 10^k \cdot y + (r_k \cdot 10^k + t)$). Ersetze q durch $q + q_k \cdot 10^k$.
  - Wenn $r_k \cdot 10^k + t$ kleiner als y ist, ist diese Zahl der gesuchte Rest und q der ganzzahlige Quotient. Ende. Sonst ersetze x durch $r_k \cdot 10^k + t$ und gehe zurück zu ∗.

- Der Algorithmus für die Division mit Rest verwendet die folgenden Rechenregeln: Summanden können vertauscht werden, Faktoren können vertauscht werden, bei mehrfacher Addition oder mehrfacher Multiplikation können Klammern weggelassen werden, man kann Herausheben und Ausmultiplizieren.
  Weiters werden der Subtraktionsalgorithmus und das Vergleichen zweier natürlicher Zahlen nach ihrer Größe verwendet.

Beispiel:
Dividiere 2023 mit Rest durch 19 !

> Wir schreiben $2023 = 20 \cdot 10^2 + 23$ (es ist $19 < 20 < 10 \cdot 19$) und dividieren 20 mit Rest durch 19: $20 = 1 \cdot 19 + 1$, daher $2000 = 100 \cdot 19 + 100$ ($k = 2, q_2 = 1$).
>
> $2023 = 100 \cdot 19 + 123$ und $123 > 16$.
>
> Wir gehen wie oben mit 123 anstatt mit 2023 weiter:
> $123 = 6 \cdot 19 + 9$ ($k = 0, q_0 = 6$)
>
> Wegen $9 < 19$ ist das Verfahren beendet und $2023 = 100 \cdot 19 + 6 \cdot 19 + 9 = 106 \cdot 19 + 9$.
> Der ganzzahlige Quotient ist 106 und der Rest ist 9.

Beachte, dass statt 106 Subtraktionen im ersten Algorithmus jetzt nur 7 Subtraktionen ausgeführt werden mussten!

Diese Vorgangsweise wird in der fachdidaktischen Literatur oft als „halbschriftliches Verfahren“ für die Division mit Rest bezeichnet, siehe (Pöll 2014) und die dort zitierte Literatur.

Das „schriftliche Verfahren“ verwendet noch eine weitere Überlegung :
Die Zahl k mit $x = s \cdot 10^k + t$, $\; y \leq s < 10 \cdot y$ und $t < 10^k$ wie oben muss nur am Anfang bestimmt werden. Wenn man auf die Bedingung $y \leq s$ verzichtet, kann man jeweils von k zu $k-1$ übergehen. Der ganzzahlige Quotient q_{k-1} kann dann auch 0 sein. Die der Reihe nach berechneten Zahlen $q_k, q_{k-1}, \ldots, q_0$ sind die Ziffern des gesuchten ganzzahligen Quotienten q.

Das schriftliche Verfahren zur Division mit Rest kann dann platzsparend so dargestellt werden:

$$\begin{array}{rrrrl}
 & 2 & 0| & 2 & 3 \;:\; 19 = 106 \\
- & 1 & 9 & 0 & 0 \\ \hline
 & & 1 & 2| & 3 \\
 & - & 0 & 0 & 0 \\ \hline
 & & 1 & 2 & 3 \\
 & - & 1 & 1 & 4 \\ \hline
 & & & & 9
\end{array}$$

oder noch kürzer

$$\begin{array}{rrrrl}
2 & 0| & 2 & 3 & \;:\; 19 = 106 \\
 & 1 & 2 & & \\
 & 1 & 2 & 3 & \\
 & & & 9 &
\end{array}$$

Ein sachgerechter Aufbau des Unterrichts der Division mit Rest ist:

- Zuerst den Grundschritt einüben: Division mit Rest mit einstelligem Quotienten.
  (Die Anzahlen der Ziffern des Divisors spielt dabei keine Rolle !)

- Dann die Idee zum Einsparen von Subtraktionen erklären.

- Allgemeiner Fall, beginnend mit einfachen Zahlen, zuerst „halbschriftlich“.

Leider legt der Lehrplan für die Volksschule etwas anderes nahe, siehe (BMBWF 2023):

3. Schulstufe: *Multiplizieren mit einstelligem Multiplikator, Dividieren durch einstelligen Divisor (ohne und mit Rest)*

4. Schulstufe: *Dividieren durch ein- und zweistelligen Divisor (ohne und mit Rest) mit sinnvollen Schwierigkeitsgraden*

Es wird außer Acht gelassen, dass bei der Division mit Rest die Anzahl der Stellen des ganzzahligen Quotienten entscheidend ist, nicht die Anzahl der Stellen des Divisors!

7. Das Bisektionsverfahren - ein Algorithmus zur Berechnung von Nullstellen stetiger Funktionen

Mit dem *Bisektionsverfahren*, das in der 10. Schulstufe unterrichtet wird, kann man die folgende Aufgabe lösen:

Gegeben sind reelle Zahlen a und b mit $a < b$, eine stetige Funktion f vom Intervall $[a,b]$ nach $\mathbb{R}$ (Menge der reellen Zahlen) so, dass $f(a) \cdot f(b) < 0$ ist ($f(a)$ und $f(b)$ haben also verschiedenes Vorzeichen) und eine positive reelle Zahl ε.
Gesucht ist ein Intervall der Länge kleiner als ε, das eine Nullstelle von f enthält. (Eine Nullstelle von f ist eine Zahl z mit $f(z) = 0$).

Am Ende dieses Abschnitts zeigen wir, dass f in $[a,b]$ mindestens eine Nullstelle hat (Zwischenwertsatz). Diese Aufgabe hat daher immer eine Lösung.

Wir verwenden die Strategie von Abschnitt 3:
Eine solche Aufgabe ist einfach zu lösen, wenn $b - a < \varepsilon$ ist. Dann ist $[a,b]$ das gesuchte Intervall.

Verändere die Aufgabe mehrfach so, dass der Definitionsbereich kleiner wird und die Funktionswerte von f an seinen Randpunkten weiterhin verschiedene Vorzeichen haben.

Sei $c := \frac{1}{2}(a+b)$ das arithmetische Mittel von a und b. Es ist $a < c < b$. Nun gibt es drei Fälle:

Wenn $f(c) = 0$, ist c eine Nullstelle von f und die Aufgabe ist gelöst.

Wenn $f(a) \cdot f(c) > 0$ ist, ersetze a durch c.

Wenn $f(a) \cdot f(c) < 0$ ist, ersetze b durch c.

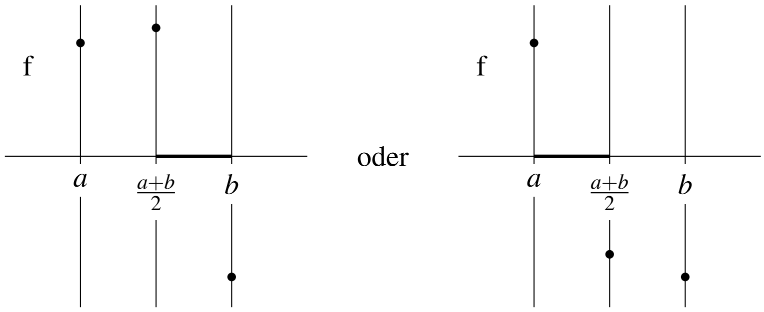

Abbildung 3: Die zwei Möglichkeiten für das neue Intervall

Das neue Intervall $[c,b]$ oder $[a,c]$ ist halb so lang wie das alte. Die Einschränkung der Funktion f auf das neue Intervall erfüllt die gleichen Bedingungen wie f. Man verändert das Intervall wie oben n-mal, wobei n so gewählt wird, dass $\frac{1}{2^n}(b-a) < \varepsilon$ ist.

Es muss noch begründet werden, dass es f (mindestens) eine Nullstelle hat.

Dazu seien $a_0 := a, b_0 := b$ und $([a_i, b_i])_{i\in\mathbb{N}}$ die Folge der wie oben konstruierten Intervalle. Die Folge $(a_i)_{i\in\mathbb{N}}$ ist monoton wachsend und durch b nach oben beschränkt, die Folge $(b_i)_{i\in\mathbb{N}}$ ist monoton fallend und durch a nach unten beschränkt. Also konvergieren beide Folgen. Wegen $b_i - a_i < \frac{1}{2^i}(b-a)$ ist $\lim_{i\to\infty}(b_i - a_i) = 0$, daher sind die Grenzwerte dieser zwei Folgen gleich. Wir bezeichnen ihn mit z. Weil f stetig ist, konvergieren die Folgen $(f(a_i))_{i\in\mathbb{N}}$ und $(f(b_i))_{i\in\mathbb{N}}$ gegen $f(z)$. Wir nehmen o. E. d. A. an, dass $f(a) > 0$ ist. Dann sind die Folgenglieder von $(f(a_i))_{i\in\mathbb{N}}$ alle ≥ 0 und die von $(f(b_i))_{i\in\mathbb{N}}$ alle ≤ 0. Daher muss $f(z) = 0$ sein.

8. Schlussbemerkungen

In den Lehrplänen für den Mathematikunterricht sind von der 2. bis zur 12./13. Schulstufe viele Algorithmen verankert, leider wird das nicht ausreichend deutlich gemacht. Um die Schülerinnen und Schüler zum algorithmischen Denken hinzuführen, ist es wichtig, diese Verfahren nicht nur einzuüben, sondern auch gut zu motivieren, zu begründen warum sie korrekt sind und die ihnen zu Grunde liegenden Strategien zu besprechen.

Eine Anregung für vertiefenden Unterricht (und auch für die Lehrplankommissionen!):
Es gibt viele Algorithmen mit Bezug zum Lebensbereich der Schülerinnen und Schüler, die in der Schule unterrichtet werden könnten, zum Beispiel der RSA-Algorithmus, der etwa bei der Verschlüsselung des PIN-Codes einer Debitkarte verwendet wird, der Algorithmus von Dijkstra zur Berechnung kürzester Wege, der Algorithmus von Prim zur optimalen Planung von gewissen Versorgungsleitungen oder der Ungarische Algorithmus für optimale Personalzuteilung, siehe dazu zum Beispiel (Clark und Holton 1994). Mit manchen dieser Algorithmen können auch Themen der reinen Mathematik im Unterricht motiviert werden, zum Beispiel sind für den RSA-Algorithmus die Themen Primzahlen und ganzzahlige lineare Gleichungen von Bedeutung.

Die Kenntnis einiger Strategien und Ideen erleichtert es, Algorithmen für weitere Probleme zu entwickeln. Für die Vermittlung von *algorithmischem Denken* ist es wichtig, im Unterricht jedes „So berechnet man das!“ mit einem „Warum berechnet man das so?“ zu verbinden. Auch die Frage „Warum interessiert man sich dafür?“ soll gestellt und diskutiert werden. Ein solcher Unterricht lässt das Selbstbewusstsein der Schülerinnen und Schüler wachsen, weil sie erkennen können: „Ich kann durch Nachdenken Probleme lösen“ (siehe dazu auch Pauer 2021).

Literatur

BMBWF (2023): *Lehrplan der Volksschule.* Stand 2023. Online: https://www.bmbwf.gv.at/Themen/schule/schulpraxis/lp/lp_vs.html

Clark, J., Holton, D. A. (1994): *Graphentheorie: Grundlagen und Anwendungen.* Heidelberg: Spektrum Akademischer Verlag.

Grcar, J. (2011): How ordinary elimination became Gaussian elimination. *Historia Mathematica 38*(2), 163 - 218.

Humenberger, H. (Hrsg.) (2017): *Das ist Mathematik 2.* Wien: öbv.

Kocher, M. (2020): Angst vor Algorithmen? Verhaltensökonomische Aspekte der Digitalisierung. Vortrag. *Lange Nacht der Forschung 9. Oktober 2020. Institut für Höhere Studien.* Wien. Online: https://www.ihs.ac.at/fileadmin/public/2016_Files/Documents/2020/langenachtderforschung_mk.pdf

Pauer, F., Stampfer, F. (2013): Primzahlen im Schulunterricht - wozu? *Schriftenreihe zur Didaktik der Mathematik der Österreichischen Mathematischen Gesellschaft 46*, 71 - 79.

Pauer, F. (2018): A computational approach to systems of linear equations. In: S. Stewart, C. Andrews-Larson, A. Berman, M. Zandieh (Hrsg.): *Challenges and strategies in teaching linear algebra. ICME-13 Monographs* (S. 299 - 316). Cham: Springer Nature.

Pauer, F. (2019): *Algebra und Diskrete Mathematik, 4. Aufl.* Skriptum. Universität Innsbruck. Online: https://www.uibk.ac.at/mathematik/personal/pauer/adm-2019/adm-2019-april-2019.pdf

Pauer, F. (2021): Persönlichkeitsbildung im Mathematikunterricht. *Schriftenreihe zur Didaktik der Mathematik der Österreichischen Mathematischen Gesellschaft 53*, 107 - 116.

Pöll, J. (2014): *Die Division mit Rest in der Primarstufe.* Diplomarbeit. Universität Innsbruck.

Salzger, B., Bachmann, J., Germ, A., Riedler, B., Singer, K., Ulovec, A. (2022): *Mathematik verstehen 2.* Wien: öbv.

Walz, G. (2023): *Das RSA-Verfahren: Verschlüsseln und Entschlüsseln auf Basis der Algebra.* Berlin: Springer Spektrum.

Ziegenbalg, J., Ziegenbalg, O., Ziegenbalg, B. (2010): *Algorithmen: Von Hammurapi bis Gödel, 3. Aufl.* Frankfurt a. M.: Verlag Harri Deutsch.

Anschrift des Verfassers

Franz Pauer

Institut für Fachdidaktik
Fakultät für LehrerInnenbildung
Universität Innsbruck
Innrain 52
6020 Innsbruck
Österreich

franz.pauer@uibk.ac.at