Der Euklidische Algorithmus - Warum nicht in der Schule?

Thorsten Holm und Wolfgang Willems

Einführung

Teilbarkeit, größte gemeinsame Teiler (ggT) und kleinste gemeinsame Vielfache (kgV) werden heute frühzeitig im Unterricht behandelt. Dabei wird der ggT mit Hilfe der Primfaktorzerlegung bestimmt. Unser Vorschlag lautet, den effizienteren, praxisrelevanten, für Schülerinnen und Schüler ebenso zugänglichen Euklidischen Algorithmus zur Bestimmung des ggT auch im Unterricht zu behandeln. Dies halten wir insbesondere für sinnvoll, da dieser Algorithmus im heutigen Computerzeitalter mehr denn je eine zentrale Rolle spielt. Es sei hier die Kryptographie (Beutelspacher1997) - die Sicherung von Daten - genannt. Dabei geht es um das Verschlüsseln von Daten, die nur für einen bestimmten Personenkreis zugänglich sein sollen. Beispiele hierfür sind per Computer an die Bank gesendete Überweisungen oder der Empfang von Fernsehsendungen, für die gezahlt werden muß. Aber auch in der Codierungstheorie - etwa beim Abspielen von Musikdaten von einer CD - wird der Algorithmus benutzt. Gegenüber einer herkömmlichen Langspielplatte werden die Musikdaten hier gegen Kratzer auf der Platte geschützt.

In der Klassenstufe 5/6 lernen Schülerinnen und Schüler die grundlegenden Begriffe *Teilbarkeit* und *Primzahlen* kennen. Der Bestimmung des ggT zweier (natürlicher) Zahlen beziehungsweise des kgV kommt dabei Bedeutung zu, vor allem in der meist anschließenden Behandlung der Bruchrechnung (Kürzen, Addition ungleichnamiger Brüche). Bei der in der Schule zur Berechnung des ggT verwendeten Methode wird zunächst die Primfaktorzerlegung der beiden Zahlen bestimmt und daraus dann der größte gemeinsame Teiler oder das kleinste gemeinsame Vielfache ermittelt. Die große Schwäche dieses Verfahrens ist natürlich, daß es nur für kleine Zahlen für Schüler praktikabel ist. Die Primfaktorzerlegung 'großer' Zahlen ist selbst mit den schnellsten Rechnern sehr aufwendig; dies ist der Grund für die Wirksamkeit des in der Praxis zur Verschlüsselung von Daten eingesetzten

RSA-Verfahrens, welches nach den drei Erfindern Rivest, Shamir und Adleman benannt ist. Wollte ein Unbefugter bei diesem Verfahren aus dem verschlüsselten Text den ursprünglichen wiedergewinnen, so müsste er aus einer etwa 150-stelligen Zahl (im Dezimalsystem), die beiden geheimgehaltenen Primzahlen ermitteln, deren Produkt die vorgegebene Zahl ist. Versuchen Sie einmal die hundertstellige Zahl

9 412 343 607 359 262 946 971 172 136 294 514 357 528 981 378 983 082 541 347 532 211 942 640 121 301 590 698 634 089 611 468 911 681

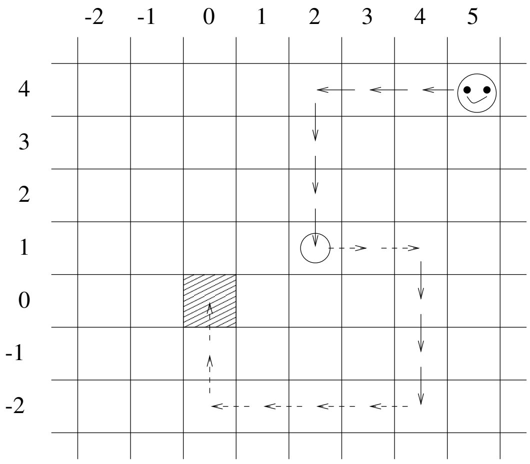
in ein Produkt von Zahlen ungleich 1 zu zerlegen. Sie werden schnell feststellen, dass dies hoffnungslos ist, auch wenn Sie gut programmieren können.

Schon das Auffinden des ggT von z.B. 899 und 1271 (welcher 31 ist) dürfte vielen Schülerinnen und Schülern mit der Faktorisierungsmethode schwer fallen. Hingegen hat der Euklidische Algorithmus den Vorteil, dass selbst große Zahlen einfach zu behandeln sind. Dabei benutzt der Algorithmus lediglich Grundbegriffe der Teilbarkeitslehre wie *Division mit Rest*. Man könnte einwenden, dass das Rechnen mit kleinen Zahlen für die Schule ausreicht. Sicher mag dies im Wesentlichen zutreffen. Andererseits, warum sollte man Schülern den besseren, effizienten und praxisrelevanten Weg vorenthalten, zumal er einfach beschreibbar ist und sich leicht an Beispielen erklären lässt?

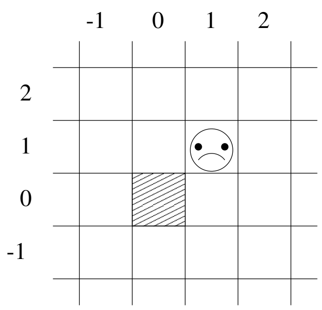
Wir wollen ausgehend von einem Spiel auf experimentellem Weg zum Begriff der Teilbarkeit kommen, die Existenz der Lösung des Spielproblems auf die Frage nach dem ggT zweier Zahlen zurückführen und schließlich mittels des Euklidischen Algorithmus Lösungen auffinden. Unser Vorschlag für die Behandlung des Euklidischen Algorithmus soll insbesondere Anregungen für die Arbeit mit besonders interessierten und begabten Schülerinnen und Schülern im Rahmen des Mathematikunterrichts oder in der außerunterrichtlichen Beschäftigung mit Mathematik (Arbeitsgemeinschaften, Schülerclubs, u.a.) vermitteln.

Problemstellung

Wir malen uns ein (im Prinzip nach allen Seiten unendliches) Schachbrett auf. Die Nummerierung der Kästchen haben wir dabei wie auf dem Achsenkreuz gewählt.



Wir starten in einem Ausgangsfeld mit einer Smiley-Figur, die sich horizontal (nach links und rechts) und vertikal (nach oben und unten) bewegen darf, und zwar entweder $a = 2$ oder $b = 3$ Schritte. Bei einem Schritt kommt dabei die Figur in ein benachbartes Kästchen. In einem Zug darf sie jedoch nur zwei oder drei Schritte in horizontaler oder vertikaler Richtung auf einmal machen. Vom Ausgangsfeld mit den Koordinaten $(5, 4)$ kommt sie also z.B. nach einem horizontalen Dreiersprung, gefolgt von einem vertikalen Dreiersprung in das mit dem Kreis bezeichnete Feld. Die Frage lautet: Kann die Figur nur mit solchen Zweier- und Dreierschritten das schraffierte Feld erreichen? Die Antwort in unserem Beispiel ist natürlich 'ja' (eine mögliche Lösung ist im Bild skizziert). Was passiert, wenn wir das Startfeld ändern, d.h. von welchen Feldern aus kann man das schraffierte Feld mit Zweier- und Dreierschritten erreichen? Die Antwort lautet: von jedem beliebigen Feld, aber das ist schon gar nicht mehr so offensichtlich. (Denn wie soll man die unendlich vielen möglichen Startfelder einheitlich behandeln?) Schwieriger wird es, wenn wir auch die erlaubten Schrittweiten a und b abändern und fragen, von welchen Startfeldern aus man mit Schritten der Längen a und b das schraffierte Zielfeld erreichen kann. Dass dies nicht für alle Ausgangspositionen möglich ist, sieht man leicht an folgendem Beispiel für die Schrittweiten $a = 2$, $b = 4$:



Egal, wie die Figur sich bewegt, sie kann nur auf Feldern landen, für die beide Koordinaten ungerade sind. Insbesondere kann das schraffierte Feld $(0, 0)$ niemals erreicht werden.

Wir werden eine allgemeine Lösung dieses Problems entwickeln. Das entscheidende zu Grunde liegende Verfahren wird der Euklidische Algorithmus sein.

Lösung des Problems - Der Euklidische Algorithmus

Wir geben uns ab jetzt Schrittweiten a und b vor. Schauen wir uns die verschiedenen Möglichkeiten zur Bewegung der Figur genauer an: Wir können z.B. zunächst a, dann b Schritte, aber auch z.B. a, nochmals a und dann b Schritte gehen. Solange wir uns nur horizontal oder nur vertikal bewegen, ist die Reihenfolge der a- und b-Schritte egal, es kommt nur auf die jeweilige Anzahl an. Machen wir r Schritte der Länge a und s Schritte der Länge b, so bewegen wir uns insgesamt $ra+sb$ Felder vom Ausgangsfeld weg. Hierbei vereinbaren wir, dass positive (negative) Zahlen r, s Bewegungen nach rechts bzw. oben (links bzw. unten) bezeichnen, entsprechend der Nummerierung der Koordinaten.

Welche ganzen Zahlen können wir überhaupt aus gegebenen a und b von der Form $ra + sb$ bilden? An Beispielen können die Schüler durch Probieren mit dieser Frage experimentieren. Für das Beispiel $a = 2$, $b = 3$ werden sie schnell vermuten, dass alle Zahlen darstellbar sind. In der Tat hat jede Zahl z die Form

$$z = z \cdot 1 = z \cdot (3 - 2) = (-z) \cdot 2 + z \cdot 3.$$

Betrachtet man das Beispiel $a = 2$, $b = 4$, so gilt dies natürlich nicht mehr, denn jede Zahl der Form $2r + 4s = 2(r + 2s)$ ist gerade.

Diese Beobachtungen an einfachen Beispielen lassen sich verallgemeinern. Dazu bezeichne $d = \text{ggT}(a, b)$ den *größten gemeinsamen Teiler* von a und b, also die größte ganze Zahl, welche a und b teilt. Wir können demnach $a = d \cdot a'$ und $b = d \cdot b'$ mit ganzen Zahlen a', b' schreiben. Jede Zahl der Form

$$ra + sb = rda' + sdb' = d(ra' + sb')$$

ist durch $d = \text{ggT}(a, b)$ teilbar. Man beachte hier die Vertauschbarkeit der Zahlen bei der Multiplikation. Für eine Lösung unseres Ausgangsproblems, vom Ausgangsfeld (x, y) das schraffierte Feld $(0, 0)$ durch horizontale und vertikale Schritte der Längen a und b zu erreichen, müssen wir uns letztendlich x Schritte horizontal und y Schritte vertikal bewegen. Welche Schritte wir zuerst machen, die horizontalen oder vertikalen, oder auch abwechselnd, spielt dabei keine Rolle. Da rein horizontale oder rein vertikale Bewegungen durch Zahlen der Form $ra + sb$ gegeben sind, erhalten wir als *notwendige* Bedingung für die Existenz einer Lösung:

$$\boxed{\text{ggT}(a, b) \mid x \qquad \text{und} \qquad \text{ggT}(a, b) \mid y}$$

In unserem Beispiel zu Beginn, mit Ausgangsfeld $(5, 4)$ und $a = 2$, $b = 3$, ist diese notwendige Bedingung erfüllt (natürlich muss dies so sein, denn wir wissen ja bereits, dass eine Lösung existiert). Wir können jetzt aber bereits leicht Beispiele angeben, in denen definitiv keine Lösung existieren kann. Nehmen wir z.B. X= $(5, 4)$ wie oben, aber als Schrittweiten $a = 6$ und $b = 2$. Nun ist $\text{ggT}(a, b) = 2$ kein Teiler von $x = 5$, also können wir das schraffierte Feld $(0, 0)$ nicht erreichen.

Es ist wichtig zu bemerken, dass eine notwendige Bedingung wie die obige noch nicht die Existenz einer Lösung garantiert. Wie aber können wir genau entscheiden, ob eine Lösung existiert oder nicht? Nehmen wir einmal an, wir können ganze Zahlen r und s finden, so dass

$$ra + sb = d = \text{ggT}(a, b).$$

Die notwendige Teilerbedingung $d \mid x$ und $d \mid y$, d.h. $x = x'd$ und $y = y'd$ mit geeigneten Zahlen x', y', liefert dann

$$(x'r)a + (x's)b = x'd = x$$

und

$$(y'r)a + (y's)b = y'd = y\ .$$

Macht die Figur also horizontal $x'r$ Schritte der Länge a und $x's$ Schritte der Länge b und vertikal entsprechend $y'r$ Schritte der Länge a und $y's$ Schritte der Länge b, so erreicht sie das Zielfeld. Wir haben somit eine Lösung gefunden.

Wie aber können wir Zahlen r, s finden, so dass

$$ra + sb = \mathrm{ggT}(a, b)$$

gilt? Dieses Problem führt uns zunächst zum Euklidischen Algorithmus, der $d = \mathrm{ggT}(a, b)$ berechnet, und dann, indem wir den Algorithmus einfach von hinten lesen, zur Bestimmung der Koeffizienten r und s (*erweiterter Euklidischer Algorithmus*). Die Koeffizienten werden häufig nach dem französischer Mathematiker Bézout (1730-1783) auch *Bézout-Koeffizienten* genannt. Grundlegend für den Euklidischen Algorithmus ist lediglich die *Division mit Rest*:
Sind a und b gegeben, so gibt es (natürliche) Zahlen q und r, wobei $0 \leq r < b$, so daß $a = q \cdot b + r$. (Hier bedeutet $r = 0$, daß a durch b teilbar ist.) Der Euklidische Algorithmus führt nun fortlaufend Divisionen mit Rest durch.

Euklidischer Algorithmus

Eingabe: $a, b \in \mathbb{N}$.

$$\begin{array}{ll}
a = q_0 b + r_1 & \text{mit } 0 \leq r_1 < b \\
b = q_1 r_1 + r_2 & \text{mit } 0 \leq r_2 < r_1 \\
r_1 = q_2 r_2 + r_3 & \text{mit } 0 \leq r_3 < r_2 \\
\quad\vdots & \quad\vdots \\
r_{i-1} = q_i r_i + r_{i+1} & \text{mit } 0 \leq r_{i+1} < r_i \\
\quad\vdots & \quad\vdots \\
r_{n-1} = q_n r_n + r_{n+1} & \text{mit } 0 \leq r_{n+1} < r_n \\
r_n = q_{n+1} r_{n+1} &
\end{array}$$

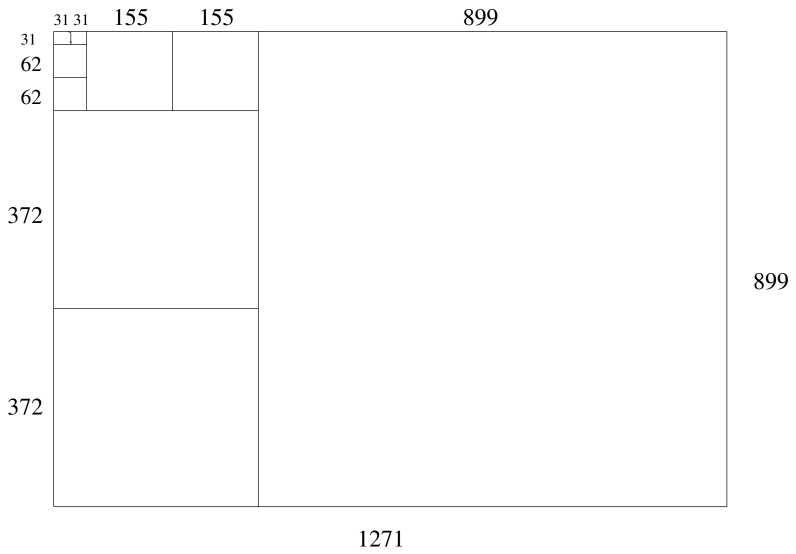
Ausgabe: $r_{n+1} = \mathrm{ggT}(a, b)$.

Natürlich werden wir uns überlegen müssen, dass der Algorithmus wirklich als Ergebnis wie behauptet den größten gemeinsamen Teiler von a und b liefert.

Aber zunächst berechnen wir als Beispiel den ggT von 1271 und 899. Wie oben bemerkt, ist die Bestimmung mit Hilfe der Primfaktorzerlegung mühsam, denn in $1271 = 31 \cdot 41$ und $899 = 29 \cdot 31$ sind die Primteiler nicht mehr ganz klein und durch die üblichen in der Schule behandelten Teilerregeln kaum aufzuspüren. Mit dem Euklidischen Algorithmus berechnet sich der ggT sehr schnell:

$$\begin{array}{rcll}
1271 & = & 1 \cdot 899 & +372 \\
899 & = & 2 \cdot 372 & +155 \\
372 & = & 2 \cdot 155 & +62 \\
155 & = & 2 \cdot 62 & +\mathbf{31} \\
62 & = & 2 \cdot 31 &
\end{array}$$

Einen visuellen Zugang zu diesem schrittweisen Verfahren bietet das Auslegen eines Rechtecks mit den Seitenlängen a und b durch möglichst große Quadrate. Das obige Beispiel sieht dann wie folgt aus.



Der Euklidische Algorithmus ist für jede Schülerin und jeden Schüler leicht durchzuführen, sobald Division mit Rest im Unterricht behandelt wurde. Der Vorteil liegt auf der Hand: Es können sehr viel grössere Zahlen behandelt werden als mit der Primfaktor-Methode.

Kommen wir zurück zur genaueren Untersuchung des Euklidischen Algorithmus und überlegen wir uns seine Korrektheit. Zwei Aspekte sind zu betrachten:

1.) (*Endlichkeit*) Endet der Algorithmus nach *endlich* vielen Schritten?

2.) (*Korrektheit*) Ist der letzte von Null verschiedene Rest r_{n+1} tatsächlich der ggT der Eingabe a und b?

Beginnen wir mit der Antwort zu 1.): Der entscheidende Punkt bei der Division mit Rest ist, dass der Rest kleiner ist als die Zahl durch die geteilt wurde. Die Folge der Reste im Euklidischen Algorithmus liefert also eine echt absteigende Folge

$$b > r_1 > r_2 > \dots r_{n-1} > r_n > r_{n+1} > 0$$

natürlicher Zahlen. Eine solche muss nach endlich vielen Gliedern abbrechen, d.h. nach endlich vielen Schritten ergibt sich bei der Division der Rest 0, wie behauptet.

Etwas komplizierter ist die Beantwortung von 2.): Erstens, ist r_{n+1} ein gemeinsamer Teiler von a und b? Dazu lesen wir die Folge der Divisionen mit Rest 'von unten nach oben': $r_n = q_{n+1}r_{n+1}$ bedeutet, dass r_{n+1} ein Teiler von r_n ist; damit teilt r_{n+1} auch $r_{n-1} = q_n r_n + r_{n+1}$, und dann auch $r_{n-2} = q_{n-1}r_{n-1} + r_n$ etc. Oben angekommen erhalten wir, dass r_{n+1} sowohl $b = q_1 r_1 + r_2$ als auch $a = q_0 b + r_1$ teilt.

Zweitens, ist r_{n+1} der *größte* gemeinsame Teiler? Sei dazu t ein gemeinsamer Teiler von a und b. Jetzt lesen wir die Divisionen mit Rest 'von oben nach oben': Die Zahl t teilt a und b, also auch $r_1 = a - q_0 b$; es folgt, dass t auch $r_2 = b - q_1 r_1$ teilt, dann auch $r_3 = r_1 - q_2 r_2$ etc.; bis endlich t auch $r_{n+1} = r_{n-1} - q_n r_n$ teilt. Insbesondere ist $t \leq r_{n+1}$ und damit r_{n+1} wie gewünscht der größte unter allen gemeinsamen Teilern von a und b.

Eine Konsequenz aus dem Euklidischen Algorithmus ist im Hinblick auf die vollständige Lösung unseres Ausgangsproblems von besonderer Bedeutung, denn es gilt:

Zu vorgegebenen natürlichen Zahlen a, b kann man ganze Zahlen r, s finden, so dass $ggT(a, b) = ra + sb$ ist.

Diese Aussage ergibt sich direkt durch 'Rückwärtseinsetzen' in den Divisionsschritten des Euklidischen Algorithmus, beginnend 'von unten'. Wir illustrieren das wieder am Beispiel $a = 1271$ und $b = 899$. Es gilt:

$$\mathrm{ggT}(1271, 899) = 31 = 155 - 2 \cdot 62$$

und durch Einsetzen der Gleichung $372 = 2 \cdot 155 + 62$ erhalten wir

$$31 = 155 - 2 \cdot (372 - 2 \cdot 155) = (-2) \cdot 372 + 5 \cdot 155.$$

Nun setzen wir den nächsten Schritt des Euklidischen Algorithmus ein:

$$31 = (-2) \cdot 372 + 5 \cdot (899 - 2 \cdot 372) = 5 \cdot 899 - 12 \cdot 372.$$

Der nächste Schritt liefert

$$31 = 5 \cdot 899 - 12 \cdot (1271 - 899) = (-12) \cdot 1271 + 17 \cdot 899$$

und wir haben die gewünschte Darstellung.

Mit Hilfe dieser Folgerung können wir nun eine vollständige Lösung unseres Problems angeben.

> Ausgehend vom Feld (x, y) können wir mit horizontalen und vertikalen Schritten der Längen a und b das schraffierte Feld $(0, 0)$ *genau dann* erreichen, *wenn* $\mathrm{ggT}(a, b) \mid x$ und $\mathrm{ggT}(a, b) \mid y$ gilt.

Es sei hier vermerkt, dass die Bézout-Koeffizienten r und s in der Darstellung $d = \mathrm{ggT}(a, b) = ra + sb$ nicht eindeutig sind, denn offensichtlich gilt auch $d = (r + b)a + (s - a)b$.

Algorithmischer Aufwand - Fibonacci-Zahlen

Für interessierte Schülerinnen und Schüler der Klassenstufen 9 und 10 kann man unsere bisherigen Beobachtungen um einen interessanten Aspekt erweitern.

Schlägt man einen Algorithmus zur Lösung eines bestimmten Problems vor, so ist eine der wichtigsten Fragen, wie 'schnell' dieser eine Lösung liefert. Im Fall des Euklidischen Algorithmus ist vor allem die Anzahl der Schritte, d.h. der durchzuführenden Divisionen mit Rest, eine entscheidende Größe und diese soll im Folgenden diskutiert werden. Natürlich wird dabei die Größe der Eingabezahlen eine Rolle spielen. Aber bei Eingaben ähnlicher Größe können durchaus sehr unterschiedliche Schrittzahlen auftreten, wie man an Beispielen leicht überprüft: Bei Eingabe von z.B. 1399 und 1057 berechnet sich der $\mathrm{ggT}(1399, 1057) = 1$ in wenigen Schritten:

$$\begin{aligned}
1399 &= 1 \cdot 1057 + 342 \\
1057 &= 3 \cdot 342 + 31 \\
342 &= 11 \cdot 31 + 1 \\
31 &= 31 \cdot 1
\end{aligned}$$

wohingegen die Eingabe 1597 und 987 zu einer recht langen Rechnung führt:

$$\begin{aligned}
1597 &= 1 \cdot 987 + 610 \\
987 &= 1 \cdot 610 + 377 \\
610 &= 1 \cdot 377 + 233 \\
377 &= 1 \cdot 233 + 144 \\
233 &= 1 \cdot 144 + 89 \\
144 &= 1 \cdot 89 + 55 \\
89 &= 1 \cdot 55 + 34 \\
55 &= 1 \cdot 34 + 21 \\
34 &= 1 \cdot 21 + 13 \\
21 &= 1 \cdot 13 + 8 \\
13 &= 1 \cdot 8 + 5 \\
8 &= 1 \cdot 5 + 3 \\
5 &= 1 \cdot 3 + 2 \\
3 &= 1 \cdot 2 + 1 \\
2 &= 2 \cdot 1
\end{aligned}$$

Die entscheidende Frage für den algorithmischen Aufwand des Euklidischen Algorithmus lautet dann für die gegebenen natürlichen Zahlen a und b:

Wieviele Schritte benötigt der Euklidische Algorithmus höchstens *zur Bestimmung des ggT von a und b?*

Anders gefragt: Bei welchen Eingaben werden, relativ zur Größe von a und b, besonders viele Divisionen mit Rest durchgeführt? (Man spricht dann vom '*worst-case*', dem für die Laufzeit des Algorithmus ungünstigsten

Fall.) Noch einmal anders gefragt: Wenn $n+1$ Divisionen benötigt werden (d.h. $\mathrm{ggT}(a,b) = r_{n+1}$), wie groß ist b (und a) dann mindestens? Dazu lesen wir wieder den Algorithmus 'von unten nach oben'. Durch Einsetzen erhalten wir schrittweise

$$\begin{array}{rcl} r_n & = & q_{n+1}r_{n+1} \\ r_{n-1} & = & q_n r_n + r_{n+1} = (q_n q_{n+1} + 1) r_n \\ r_{n-2} & = & q_{n-1} r_{n-1} + r_n = (q_{n-1} q_n q_{n+1} + q_{n-1} + q_n + 1) r_n \\ \vdots & & \vdots \end{array}$$

Schließlich ist b von der Form $b = Ar_n$, wobei A eine Summe von Produkten der auftretenden $q_1, \ldots, q_{n+1}$ ist. Da alle q_i positive ganze Zahlen sind, wird b minimal, wenn $r_{n+1} = 1$ ist und alle $q_i = 1$ sind. In diesem Fall ergibt sich für die im Euklidischen Algorithmus auftretenden Reste durch Einsetzen schrittweise

$$\begin{array}{rcl} r_n & = & 1 \\ r_{n-1} & = & r_n + r_{n+1} = 2 \\ r_{n-2} & = & r_{n-1} + r_n = 3 \\ r_{n-3} & = & r_{n-2} + r_{n-1} = 5 \\ \vdots & & \vdots \\ r_{i-1} & = & r_i + r_{i+1} \\ \vdots & & \vdots \end{array}$$

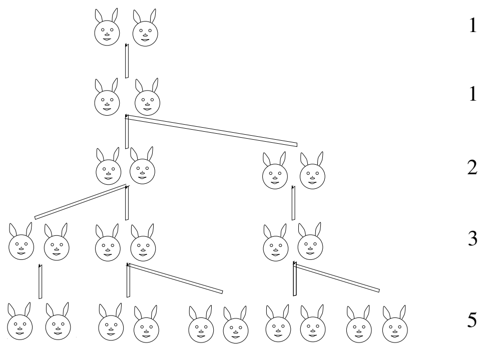
Jeder Rest ist also die Summe der beiden folgenden Reste; die entstehende Zahlenfolge taucht an vielen Stellen der Mathematik auf und ist benannt nach Leonardo di Pisa, genannt Fibonacci (Filius Bonacci), der von etwa 1170 bis circa 1250 lebte. Die *Fibonacci-Zahlen* $(F_n)_{n\in\mathbb{N}}$ sind definiert durch

$$F_1 = 1 \ , \quad F_2 = 1 \ , \quad F_{j+1} = F_j + F_{j-1} \ \text{ für } j \geq 2.$$

Die ersten Folgenglieder lauten also

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597,...

Diese Folge trat 1202 in Fibonaccis Buch *Liber Abbaci* als Übungsaufgabe auf, in welcher die Nachkommenschaft eines Kaninchenpaares zu zählen war. Dabei traf Fibonacci die Annahme, dass ein Kaninchenpaar vom zweiten Monat seiner Existenz an jeden Monat ein neues Paar hervorbringt, welches sich in gleicher Weise fortpflanzt, und fragt dann nach der Anzahl der Kaninchen nach einem Jahr.



Die kleinste Eingabe b, für die der Euklidische Algorithmus genau $n+1$ Schritte benötigt, ist also $b = F_{n+1}$ (und dann $a = F_{n+2}$). Übrigens ist jetzt auch klar, warum in obigem Beispiel bei der Eingabe $F_{17} = 1597$ und $F_{16} = 987$ recht viele Schritte erforderlich waren. Wir können also unsere Laufzeitüberlegungen zusammenfassen zu:

Die Fibonacci-Zahlen liefern den 'worst-case' für den euklidischen Algorithmus.

Eine zufriedenstellende Antwort auf die Frage nach dem algorithmischen Aufwand des Euklidischen Algorithmus liefert das folgende Resultat von Lamé (1844), welches eine von a und b abhängende obere Schranke für die Anzahl der benötigten Schritte angibt. Ist $a \geq b$, so besagt es:

Der Euklidische Algorithmus benötigt zur Berechnung von ggT(a,b) höchstens $5k$ Divisionen mit Rest, sofern b genau k Stellen im Dezimalsystem hat.

Die Abschätzung erhält man, wenn man das Anwachsen der Fibonacci Zahlen genauer untersucht (Schwarz1998).

Die Fibonacci-Zahlen spielen nicht nur in vielen Bereichen der Mathematik eine Rolle, sondern auch an vielen Stellen in der Natur, etwa in der Anordnung von Sonnenblumenkernen, in Pinienzapfen,.... Für mehr Informationen ist ein Besuch der Internet Seite
`http://www.ee.surrey.ac.uk/Personal/R.Knott/Fibonacci/fib.html`
lohnenswert.

Historisches

Wie der Name bereits sagt, geht der Euklidische Algorithmus auf Euklid (ca. 320 - 275 v.Chr.) zurück. Er befindet sich im siebten Buch der *Elemente* im Paragraph 2 (*Die Elemente von Euklid*, 3. Teil, Buch VII-IX, Akademische Verlagsgesellschaft Leipzig, 1935). Ein Algorithmus zur Lösung der Gleichung

$$ax + by = c$$

im Bereich der ganzen Zahlen wurde erstmals von Àryabhata (476-550) in seinem in Sanskrit geschriebenen Buch Àryabhatíya über Astronomie gegeben. Die allgemeine Berechnung der Koeffizienten r und s mittels des erweiterten Euklidischen Algorithmus in

$$ra + sb = \mathrm{ggT}(a, b)$$

geht auf Euler (1707-1738) zurück. Untersuchungen über den Aufwand zur Berechnung des ggT gab es bereits im 19. Jahrhundert, u.a. von Jacques Binet (1841) und Gabriel Lamé (1844). Hier wird der Zusammenhang zu den Fibonacci-Zahlen im 'worst-case' aufgezeigt.

Übungsbeispiele

Im Internet befinden sich viele Web-Seiten über Euklid und seinen Algorithmus. Z.B. sind unter

`http://cut-the-knot.com/blue/Euclid.html`

viele Informationen gesammelt. Den ggT und die Bézout-Koeffizienten zweier Zahlen kann man unter

`http://users.erols.com/eweidaw/applets/EuclidExtension.html`

interaktiv berechnen lassen.

Sehr zu empfehlen ist das an Schülerinnen und Schüler, aber auch an Lehrerinnen und Lehrer gerichtete Buch von Bartholomé, Rung und Kern. Fast 80 Seiten sind Themen rund um den Euklidischen Algorithmus gewidmet. Zudem haben die Autoren viele schöne Aufgaben gesammelt.

Das folgende Arbeitsblatt mit Aufgaben möge zur Übung dienen.

Rund um den Euklidischen Algorithmus

1. Bestimme den ggT und das kgV von 24 und 18 und multipliziere die beiden Zahlen. Vergleiche dies mit dem Produkt aus 24 und 18. Welche Vermutung legt dies nahe? Kontrolliere andere kleine Beispiele.

2. Bestimme den ggT von 6150 und 360 mittels des Euklidischen Algorithmus und berechne das kgV nach der Vermutung aus 1. Kontrolliere das Ergebnis anhand der Primfaktorzerlegung.

3. Bestimme den ggT von 131071 und 524287.

4. Berechne ganze Zahlen x und y, so daß $x \cdot 11 + y \cdot 7 = 1$.

5. Kann man mit zwei Sanduhren ein 7-minütiges Ei kochen, sofern die eine 5 Minuten und die andere 4 Minuten läuft?

Literatur

Bartholomé, A., Rung, J. und Kern, H.: Zahlentheorie für Einsteiger,Vieweg 1996.
Beutelspacher, A., Geheimsprachen, Verlag C.H.Beck, 1997.
Schwarz, F.: Einführung in die Elementare Zahlentheorie, Vieweg 1998.

Dr. Thorsten Holm
Prof. Dr. Wolfgang Willems
Otto–von–Guericke–Universität Magdeburg
Institut für Algebra und Geometrie
Postfach 4120
39016 Magdeburg