

# Informatik fürs Leben lernen

Till Tantau<sup>1</sup>

**Abstract:** Eine Debatte um die Frage, welche Inhalte und Themen die informatische Bildung an Schulen umfassen muss, sollte weit nach vorne blicken: Weniger die aktuellen Trends in der Wirtschaft oder Forschung erscheinen richtungsweisend, sondern was in 25 oder noch mehr Jahren immer noch oder wieder relevant ist. Ein solch weiter Blick in die Zukunft der Informatik ist natürlich ausgesprochen schwierig, da sich die Informatik, seit sie existiert, in einem ständigen Umbruch befindet. Er soll hier trotzdem versucht werden und daraus verschiedene Konsequenzen für die mögliche Gestaltung und Entwicklung des Informatikunterrichts in der Gegenwart abgeleitet werden. Einerseits ergibt sich, dass es mehrere „zeitlose“, grundlegende und spezifisch informatische Ideen gibt, die dauerhaft Teil der allgemeinen Bildung sein sollten. Andererseits sind die typischen Gegenstände der Wissenschaft Informatik – wie Code oder Systeme oder Protokolle – immer weniger für sich stehende, rein technische Artefakte, sondern vielmehr in komplexe soziale Kontexte eingebettete Produkte menschlicher Kollaboration. Neben einem prinzipiellen Verständnis technischer Aspekte wird daher das Wissen um die Entstehung, Entwicklung und Nutzung von Code oder Systemen durch Menschen zentraler Bestandteil einer auf die Zukunft ausgerichteten informatischen Bildung sein.

**Keywords:** Allgemeinbildung; Kollaboration; Kodierung; Algorithmen; Grenzen

## 1 Einleitung

In fünf Jahren wird die erste Empfehlung der Gesellschaft für Informatik zum Schulfach Informatik, die »Zielsetzungen und Inhalte des Informatikunterrichts« [Br76], ihr 50-jähriges Jubiläum feiern. Seitdem gab es etwa einmal pro Dekade einen neuen Aufruf: 1992 die »Dagstuhler Empfehlung zur Aufnahme des Fachs Informatik in den Pflichtbereich der Sekundarstufe II« [Bu92], gefolgt 2004 von der »Zweite[n] Dagstuhler Empfehlung zur Aufnahme des Fachs Informatik in den Pflichtbereich der Sekundarstufe I« [Br04] und schließlich 2016 das »Dagstuhl Manifesto« mit der »Dritte[n] Dagstuhl-Erklärung zur Informatischen Bildung in der Schule« [Ge15]. Die Titel verraten dabei schon, dass der Anspruch an die Bedeutung und Breite von Informatikunterricht beständig gestiegen ist. Dem steht gegenüber [Ri20, Tabelle 1], dass im Jahr 2020 Informatikunterricht in der Sekundarstufe I in nur einem Bundesland durchgehend verpflichtend ist; in vier weiteren Bundesländern sind minimale verpflichtende Anteile in der Sekundarstufe I vorhanden und in zwei Bundesländern ist Informatik in keiner Weise vorgesehen – weder verpflichtend, noch als Wahlfach, noch integriert in andere Fächer.

Die Informatik hat es in den letzten 50 Jahren nicht geschafft, als Disziplin wahrgenommen zu werden, der eine gleichberechtigte Aufnahme in den Kanon der Schulfächer neben

---

<sup>1</sup> Universität zu Lübeck, tantau@tcs.uni-luebeck.de

Mathematik, Physik, Kunst oder Sport gebührt. Dies erscheint zunächst unverständlich, durchdringt die Informatik in Form der Digitalisierung doch schon seit längerer Zeit in offenkundiger Weise sämtliche Lebensbereiche; die Informatik wird schon als vierte Kulturtechnik auf derselben Stufe gesehen wie Lesen, Schreiben und Rechnen. Dies sind starke Argumente, jedoch könnte man auch für andere Disziplinen mit einigem Recht verlangen, dass sie ordentliche Schulfächer sein sollten, da sie für die zukünftige Lebenswirklichkeit der jetzigen Schüler:innen eine ebenso große Bedeutung haben werden: Medizin und Gesundheit, Wirtschaft und Finanzen, Recht und Verträge. Im Kampf um knappe Stunden in den Schulcurricula kann weder die prinzipielle noch die aktuelle Bedeutung einer Disziplin hinreichend dafür sein, dass sie zum verpflichtenden Schulfach geadelt wird.

Es gibt natürlich viele Gründe, weshalb das Schulfach Informatik nun dort steht, wo es steht: Fundamentale Eingriffe in die Schulcurricula (und nichts anderes ist die Einführung eines neuen Schulfachs von der Sekundarstufe I an mit Anspruch auf Teilhabe schon im Primarbereich) verdienen Vorsicht und müssen gut durchdacht sein. Insbesondere muss es überhaupt Lehrkräfte für das Fach geben, was gerade in der Informatik zu einem Henne-Ei-Problem führte. Schließlich müssen Schulcurricula den Schüler:innen Wissen und Kompetenzen mit auf den Lebensweg geben, die ihnen später und langfristig nützlich sind. Dies erscheint gerade bei der extrem schnelllebigen Informatik schwierig – Curriculumsverantwortliche fragen sich eher, welche Inhalte des Informatikunterrichts noch relevant sein werden, wenn das Abschlusszeugnis überreicht wird.

**Beiträge dieser Arbeit.** Ziel dieses Artikels ist es, Aspekte der Informatik zu identifizieren, die auch in 25 Jahren, in 50 Jahren und in 100 Jahren noch von Relevanz für die jetzigen Schüler:innen oder deren Kinder sein werden. »Zeitlose« Aspekte gibt es in der Informatik natürlich zuhauf: Man kann mit einigem Recht sagen, dass mein Forschungsgebiet – die Theoretische Informatik – sich ausschließlich mit den zeitlosen Aspekten der Informatik beschäftigt. Weniger klar ist aber, ob – und wenn ja welche – dieser Aspekte man wirklich »fürs Leben lernen« sollte: Die Umwandlung von nichtdeterministischen in deterministische Automaten sicherlich nicht.

Jeder der Abschnitte dieser Arbeit wird eine spezifische »Idee« der Informatik ausarbeiten, von der ich der Meinung bin, dass sie fundamentale, eigenständige Einsichten in das »Wesen der Dinge« erlaubt. Dies ist eine zugegebenermaßen etwas pathetische Formulierung – jedoch erscheint mir dieser Aspekt wichtig und weder in gängigen Debatten zur Bedeutung der Informatik noch in schulischen oder hochschulischen Lehrmaterialien reflektiert. So wird in Abschnitt 2 die Erkenntnis vorgestellt, dass sich alle Kommunikation als Folge von Nullen und Einsen darstellen lässt, und ich werde argumentieren, dass dies *weit* mehr ist als Lemma 42 der Kodierungstheorie. Vielmehr steht dies auf einer Stufe mit der Erkenntnis, dass alle Materie aus Atomen aufgebaut ist. Es handelt sich um eine *paradigmatische* Erkenntnis, die auch mit der entsprechenden Verve und Breite im Unterricht vorgetragen

werden kann und soll, da sie in Form eines Paradigmas Schüler:innen eine (neue, andere) Sicht auf »die Welt« bietet.

**Verwandte Arbeiten.** Bereits 1994 findet sich im *Bulletin of the EATCS* ein Aufsatz [Sc94] mit dem schönen Titel »Fundamental Ideas of Computer Science«, der fundamentalen Ideen der Informatik für den (hoch)schulischen Unterricht nachspürt. Im Vergleich zur vorliegenden Arbeit liegt der Fokus dort auf den fachinternen Fundamenten der Informatik (wie dem für die Informatik in der Tat zentralen Konzept der »Übersetzung«, nach dem eine algorithmische Idee verschiedene sprachliche Stufen durchläuft, um am Ende als Darstellung auf Gatterebene zu enden). Im Folgenden hingegen werden fundamentale Ideen in den Blick genommen, die gerade auch außerhalb der Informatik von hoher Relevanz sind, die aber originär aus der Informatik stammen.

Andere Disziplinen verfügen bereits über Sammlungen von »Basisideen«, die auch jenseits des speziellen Fachs Wirkung entfalten, siehe die Sammlungen beispielsweise in der Physik in den Empfehlungen der Kultusministerkonferenz [Ko05, Abschnitt 2.1] und auch die Ergebnisse der Studie Deutschen Physikalischen Gesellschaft [GH14] von 2014. Für die Informatik sucht man 2021 auf den Seiten der Kultusministerkonferenz hingegen vergeblich nach Bildungsstandards, ebenso auf den Seiten des Instituts für Qualitätsentwicklung im Bildungswesen. Ein konkreter Anlass für erste Ideen zur Liste in dieser Arbeit entstand im Kontext der *Creative Mathematical Sciences Communication Conference 2016* [Re16].

## 2 Alle Kommunikation lässt sich als Folge von Nullen und Einsen darstellen

*Nur scheinbar hat ein Ding eine Farbe, nur scheinbar ist es süß oder bitter; in Wirklichkeit gibt es nur Atome und leeren Raum.*

Demokrit, nach [Ca35, Seite 135]

Es gibt Erkenntnisse, die, wenn man sie verstanden, akzeptiert und verinnerlicht hat, im wahrsten Sinne des Wortes den Blick auf die Realität ändern. Es macht für Sie beim Betrachten dieser Zeilen einen Unterschied, ob Sie darum wissen, dass die Welt »aus Atomen aufgebaut ist«, dass die Buchstaben nach Demokrit »nur scheinbar eine Farbe« haben. Mag dies im Alltag unwichtig erscheinen, so ist doch der Blick auf Gegenstände und erst recht auf unsichtbare Dinge wie Luft ein ganz anderer, als wenn man keine Vorstellung davon hat, woraus Gegenstände letztendlich eigentlich »gemacht« sind. Dabei ist es nicht nötig, ein tiefes Verständnis des aktuellen Standardmodells der Teilchenphysik zu haben – allein die zentrale Idee »alles besteht aus Atomen« ist eine eigene paradigmatische Sicht auf die Welt, die jedem Menschen zumindest einmal aufgezeigt werden sollte. Dieses Wissen kann und sollte dann noch vertieft werden, beispielsweise um das Konzept des chemischen

Elements oder die Arten der Atombewegung – dies festigt und erklärt aber lediglich das zugrundeliegende Paradigma.

Fundamentale Erkenntnisse dieser Art gibt es in der Physik viele und sie werden zu Recht in der Schule gelehrt: Ideen wie die Schwerkraft oder auch – zugegebenermaßen als sehr fortgeschrittene Idee – der Umstand, dass Zeit nicht absolut, sondern relativ ist. Die Physik ist aber nicht die einzige Quelle: Ebenso ist die Idee, dass alle Organismen aus Zellen aufgebaut sind und dass die meisten Krankheiten von sich vermehrenden, mikroskopischen Organismen ausgelöst werden, von solch fundamentaler Natur, dass man die Welt unterschiedlich wahrnimmt und zu anderen alltäglichen Einschätzungen und Entscheidungen kommt, wenn man diese Ideen nicht kennt.

Ich möchte im Folgenden nun argumentieren, dass der Satz »alle Kommunikation lässt sich als Folge von Nullen und Einsen darstellen« eine ebenso fundamentale Erkenntnis darstellt wie »alle Materie ist eine Ansammlung von Atomen«.

In Mathematik und Philosophie reicht die Idee, sprachliche Ausdrücke mit Hilfe von Zahlen zu kodieren, weit zurück. So schrieb Leibniz bereits 1666 (ohne freilich seinem Wunsch und Anspruch auch nur entfernt gerecht zu werden):

Quando orientur controversiae, non magis disputatione opus erit inter duos philosophos, quam inter duos Computistas. Sufficiet enim calamos in manus sumere sedereque ad abacos, et sibi mutuo (accito si placet amico) dicere: *calculemus*.<sup>2</sup>

Gottfried Wilhelm Leibniz, nach [Ge90, Seite 200]

George Boole steuerte 1854 dann die Idee bei [Bo54], logische Folgerungen und Kodierung auf Nullen und Einsen zu stützen. Am konsequentesten durchgeführt worden ist die Übersetzung von mathematischen Sachverhalten in Zahlen wohl in Form der »Gödelisierung«, also der Idee von Kurt Gödel von 1930 im Kontext seines Beweises des Unvollständigkeitssatzes [Gö31], eine mathematische Formel als Zeichenkette aufzufassen und diese wiederum als Zahl zu kodieren (nämlich durch die Kodierung der verschiedenen Zeichen als Potenzen in der Primfaktordarstellung der Zahl).

Trotz dieser langen Historie (welche man bei Bedarf noch viel weiter zurückverfolgen und verbreitern könnte) ist die Umwandlung von mathematischen Texten in Zahlen aus mathematischer Sicht doch eher eine umständliche technische Notwendigkeit und ihre Verwendung innerhalb des Gödel'schen Unvollständigkeitssatzes schon fast eine Skurrilität.

---

<sup>2</sup> Etwa: »Wenn sich eine Meinungsverschiedenheit ergibt, so wird nicht mehr Diskussion zwischen zwei Philosophen nötig sein als zwischen zwei Rechen-Spezialisten. Es reicht dann nämlich, die Stifte in die Hand zu nehmen, sich an die Rechentische zu setzen, und sich einander (wenn man möchte, nach Hinzuziehung eines Freundes) zu sagen: *Rechnen wir es aus!*«

Eine andere Disziplin neben der Mathematik, die sich schon von ihrem Selbstverständnis her mit der Frage der Kodierung von Kommunikation beschäftigt, ist natürlich die Linguistik beziehungsweise die Sprachwissenschaft ganz allgemein. Begriffe wie das semiotische Dreieck – ob man es nun schon bei Aristoteles verortet oder erst bei Ogden und Richards [OR23] – sind hier zentral und prägen unsere Vorstellung von der Bedeutung sprachlicher Ausdrücke. Jedoch steht die verbale menschliche Kommunikation im Zentrum der Linguistik und von Fächern wie Deutsch oder Englisch. Zu der Frage, ob (und wenn ja, wie) man einen 3D-Film kodieren könnte, können, wollen und sollen diese Fächer keine Aussage machen.

Erst durch die Informatik entsteht aus der Idee der Kodierung von beliebiger Information als Folge von Nullen und Einsen etwas völlig Neues: die Erkenntnis, dass schlicht alle menschliche und nichtmenschliche Kommunikation prinzipiell als langer Strom von Nullen und Einsen dargestellt werden kann. Wir können ein Gedicht ebenso als Bitstring darstellen wie ein Bild, ein Video, ein Spiel, einen Konzertmitschnitt, das 3D-Modell eines Wolkenkratzers oder die Messdaten einer Weltraumsonde. Null und Eins sind die Atome der Information, aus denen sich schlicht alles – nach anfangs recht einfachen, später immer komplexeren Regeln – zusammensetzen lässt.

Die Tragweite der Idee »alle Kommunikation lässt sich als Folge von Nullen und Einsen darstellen« geht aus meiner Sicht und Erfahrung weit über die schlichte Tatsache hinaus, dass man eben Zahlen und Buchstaben als Bitstrings kodieren kann. Lehrreich, wenn auch anekdotisch, ist dabei zunächst eine persönliche Erfahrung aus meiner Studienzeit. Eingeschrieben in die Studiengänge Informatik und Mathematik brachte ich zu Vorlesungen zur axiomatischen Mengenlehre mein aus der Informatik stammendes Verständnis für die »selbstverständliche Kodierbarkeit aller Dinge« mit. Damals irritierend, in der Rückschau nun faszinierend, war zu sehen, in welcher epischer Breite über mehrere Doppelstunden Beweise zu Sätzen der axiomatischen Mengenlehre vorgestellt wurden, die sich aus Informatik-Sicht zusammenfassen lassen als »Formeln sind Zeichenketten«, »Zeichenketten kann man als Bitstrings schreiben« und »Bitstrings kann man hintereinander schreiben und erhält wieder Bitstrings«. Entscheidend erscheint mir, dass mein damaliger Mathematikprofessor und ich *keine geteilte Sicht auf das Wesen einer Formel* hatten: War für ihn eine Formel ein mathematisches Objekt mit einer komplexen inneren Struktur, das nur sehr mühselig von einer formalen Ebene auf eine andere transferiert werden kann, erschien (und erscheint) mir völlig klar, dass Formeln wie »eben alles andere auch« ohne weiteres kodiert und dekodiert werden können.

Wenn schon ein Mathematiker, dessen Forschungsgebiete formale Logik und Mengenlehre sind, nicht die Vorstellung der universellen Kodierbarkeit aller Dinge verinnerlicht hat, wie überraschend, neu und erhellend kann sie dann für Schüler:innen oder auch Studierende sein? Seit nunmehr 16 Jahren halte ich eine Vorlesung »Einführung in die Informatik«, die mit einem Kapitel zu Information und Daten beginnt. Über einzelne Kodierungsschritte, beginnend mit Zahlen, Texten, Pixeln, Bildern und dann Videos wird erarbeitet, wie ein Film zu einer Folge von Nullen und Einsen wird, die sich auf eine Scheibe brennen lässt

oder über eine Datenleitung verschickt werden kann. Die Reaktionen und Nachfragen zeigen immer wieder, welch ein »Augenöffner« diese Idee für viele Studierende ist.

Man möchte an dieser Stelle nun vielleicht einwenden, dass die Behauptung »alle Kommunikation lässt sich als Folge von Nullen und Einsen darstellen« einer einfachen Realitätsüberprüfung nicht standhält: Zwischenmenschliche Kommunikation verläuft offenbar *nicht* nur auf der Ebene eines gut kodierbaren, verbal vorgetragenen Textes, vielmehr spielt neben Faktoren wie der Stimmlage und Intonation die nonverbale Kommunikation in Form von Körperhaltung und Gestik eine entscheidende Rolle – selbst haptische Elemente in Form von Berührung oder Fühlen können wesentliche Bestandteil von Kommunikation sein.

Auf diesen Einwand möchte ich zwei Antworten geben: Zum einen sind eben nicht nur gesprochene Texte kodierbar, sondern ebenso der genaue Ton, ein Video des Gegenübers, ob in zwei oder drei Dimensionen, und selbst eine haptische oder olfaktorische Erfahrung ist prinzipiell kodierbar. Zum anderen – und wichtiger – wird erst durch das Verständnis der allgemeinen Kodierbarkeit von Kommunikation klar, warum wir beispielsweise »überhaupt etwas bei einer Video-Konferenz im Vergleich zu einem Präsenztreffen verlieren«: Es fehlen uns genau die Anteile der Kommunikation, die sonst auf der aktuell nicht kodierten Ebene ablaufen. Hierfür ein tieferes Verständnis zu vermitteln, erscheint mir als wichtiges und würdiges Ziel von schulischem Unterricht.

### 3 Algorithmen

Die Überschrift dieses Abschnitts lautet einfach nur »Algorithmen«, da die Idee des Algorithmus »an sich« und ohne konkrete Einsatzszenarien von fundamentaler Bedeutung ist. Es gibt viele weitere Ideen der Informatik rund um Algorithmen – wie den Umstand, dass es universelle Algorithmen gibt, die also jeden beliebigen anderen Algorithmus simulieren können, oder dass Algorithmen mit Datenstrukturen interagieren – jedoch ist es zunächst schon das Wissen um die reine Existenz von Algorithmen, das die Sicht auf die Umwelt verändert.

Die Idee des Algorithmus ist sehr alt und wir finden Algorithmen in den altertümlichen oder mittelalterlichen Schriften sowohl beim Gelehrten Abu Dscha'far Muhammad ibn Musa al-Chwārizmī (dessen Herkunftsangabe »al-Chwārizmī«, also »aus Choresmien«, Ausgangspunkt der Etymologie des Wortes »Algorithmus« ist) wie bei Adam Riese. Eine wichtige frühe Algorithmikerin ist die Countess of Lovelace Augusta Ada King-Noel, deren faszinierende Abhandlung betreffend Algorithmen für die *Analytical Engine* von Charles Babbage zur Berechnung von Bernoulli-Zahlen bereits 1843 erschien [Me43, der Anhang]. Insofern erscheint das Konzept des Algorithmus zunächst der Mathematik zu entspringen als ein eher klassisches Werkzeug, den Ablauf von Berechnungen exakt zu beschreiben. Die Notwendigkeit, Algorithmen wirklich zu »verstehen«, wurde Anfang des 20. Jahrhunderts im Kontext des Hilbert'schen Programms dringlich, dessen Ziel es war (vereinfacht gesprochen), »mathematische Wahrheit auszurechnen«.

Man tritt den großen Geistern der Zeit in diesem Kontext, insbesondere Rózsa Péter, Wilhelm Ackermann, David Hilbert, Alonzo Church oder Kurt Gödel, nicht zu nahe, wenn man festhält, dass ihr Versuch, den Begriff des »Berechnens« oder allgemein des »algorithmischen Vorgehens« formal zu fassen, wenig erfolgreich war – genauer war er nur »innermathematisch« erfolgreich. Das Endresultat der damaligen Bemühungen um eine Definition von »berechenbar« war das Konzept der  $\mu$ -Rekursion – und dieses ist selbst aus heutiger Sicht ein eher unhandliches Werkzeug mit doch recht schmalen Anwendungsbereichen in der Rekursionstheorie.

Es war erst die Informatik, beginnend mit dem bahnbrechenden Aufsatz Alan Turings [Tu37] von 1937, die den Begriff des Algorithmus so ausgeweitet und weiterentwickelt hat, dass etwas qualitativ Neues entstand. Dies wurde auch von der mathematischen Gemeinde schnell erkannt, zuallererst von Alonzo Church selbst, jedoch hat sich der Algorithmusbegriff seitdem von seinen rein mathematischen Ursprüngen aus meiner Sicht vollständig emanzipiert.

Algorithmen bestimmen heutzutage im wahrsten Sinne des Wortes das Verhalten der meisten Geräte in unserer Umwelt. Dies schließt bei vielen Menschen und sicherlich bei den allermeisten Schüler:innen Geräte ein, die sie – wie ein Handy – ein Großteil ihres Lebens in unmittelbarer Reichweite haben. Es erscheint mir wahrscheinlich, dass die heutigen Schüler:innen eine Welt erleben werden, in der Algorithmen nicht nur in vielen Geräten stecken werden, sondern in praktisch allen Gegenständen inklusive Kleidung, Stühle, Fenster, Wände, ja vielleicht sogar in anderen Lebewesen. Zu wissen, dass das erlebte Verhalten der eigenen Umwelt zu einem großen und immer größer werdenden Teil von Algorithmen bestimmt wird, führt zu einer ganz konkret anderen »Sicht auf die Umwelt« als ohne dieses Wissen.

Etwas fortgeschrittener, aber trotzdem noch von fundamentaler Natur, ist die Erkenntnis, dass komplexere Algorithmen eine völlig eigenständige Kategorie geistiger Schöpfungstätigkeit darstellen. Sie scheinen mir am ehesten vergleichbar mit den Beweisen für mathematische Sätze: Ähnlich wie Beweise in einem mühevollen Prozess sowohl *erfunden* wie *gefunden* werden müssen, müssen auch Algorithmen gleichermaßen erfunden und gefunden werden. So wie die Mathematik zu recht darauf besteht, dass im Rahmen des Mathematikunterrichts Schüler:innen zumindest erkennen, was ein Beweis überhaupt ist und warum man einen solchen überhaupt braucht, so sollten sie auch wissen, was ein Algorithmus ist und wofür man ihn braucht.

## 4 Viele Probleme werden sich nie mit Computern lösen lassen

Die Erkenntnis, dass es überhaupt irgendwelche Probleme gibt, die sich *nicht* – zumindest mittelfristig – mit Computern lösen lassen, erscheint in deutlichem Widerspruch zur Erfahrungswirklichkeit vieler Menschen und insbesondere der heutigen Schüler:innen: Jedes Jahr verschiebt sich die Grenze dessen, was von Computern erledigt werden kann und wird. Was gestern noch eher unter Science-Fiction verbucht wurde, ist heute Alltag: Wir können im

wahrsten Sinne des Wortes »mit Maschinen reden«<sup>3</sup>; wir erhalten Sonnenstundenvorhersagen für die nächsten Stunden mit einer Exaktheit, dass man die Uhr danach stellen kann; Algorithmen übersetzen Texte von einer Sprache besser in eine andere, als Menschen dies können; die Qualität von Krebsdiagnosen wird deutlich erhöht, wenn man statt erfahrenen Patholog:innen lieber Algorithmen befragt; Artikel werden von Algorithmen geschrieben, die denjenigen von Journalist:innen in nichts nachstehen; – und so weiter und so fort.

Zu dieser alltäglichen Wahrnehmung eines scheinbar grenzenlosen Potentials und einer scheinbar grenzenlosen Entwicklungsmöglichkeit von Computern, Algorithmen und Codes kommen dann noch die Erzählungen des Mainstream-Kinos hinzu (aber auch von avantgardistischen Science-Fiction-Romanen), die Computern im Rahmen einer »Singularität« (oder einfach ohne Begründung) wahrhaft phantastische Fähigkeiten andichten.

Vor diesem Hintergrund haben Menschen ohne das Wissen um die prinzipiellen und praktischen Grenzen von Algorithmen eine andere Wahrnehmung der Welt als solche mit diesem Wissen und sie werden andere Einschätzungen und Entscheidungen treffen.

**Die theoretischen Grenzen.** Dass Computer nicht alles können, zeigt Alan Turing [Tu37] in seinem berühmten, bereits erwähnten Aufsatz »On Computable Numbers with an Application to the Entscheidungsproblem« bereits 1937 und damit lange bevor es überhaupt reale Computer im heutigen Sinne gab. Er zeigt dort, dass das Halteproblem unentscheidbar ist. Danach wurden weitere Probleme gefunden, die formal unentscheidbar sind (also unentscheidbar im strengen Sinne der Theoretischen Informatik) wie das im Vergleich zum Halteproblem deutlich eingängigere Post'sche Korrespondenzproblem [Po46]. Jedoch wird nun selbst innerhalb der Informatik die Unentscheidbarkeit von konkreten Problemen oft als Kuriosität angesehen ohne Bedeutung für praktische Fragestellungen und gerne abgeschoben an die Vorlesung zur Theoretischen Informatik.

Dem steht aus meiner Sicht gegenüber, dass es von grundlegender Bedeutung ist, ob es überhaupt Dinge gibt, »die Computer nicht können«. Insbesondere kann die eingangs vorgestellte Liste der immer neuen von Computer gelösten Probleme nicht beliebig verlängert werden. Dies zu wissen, mag manche Menschen beruhigen, manche enttäuschen – auf jeden Fall sollten sie es wissen. Die durchaus verbreitete Vorstellung von (und Angst vor) einer »Singularität« mit einer hyperbolisch ins Unendliche ansteigenden Rechenkraft einer oder mehrerer Maschinen, die dann gottgleiche Fähigkeiten entwickeln, ist schlicht eins: Quatsch.

**Die praktischen Grenzen.** Mindestens genauso wichtig wie das Wissen um die theoretischen Grenzen des Berechenbaren ist das Wissen darum, dass nicht jedes berechenbare

---

<sup>3</sup> So habe ich mich kürzlich darüber geärgert, dass ich einem wenige Jahre alten Gerät nicht einfach sagen konnte, welchen Film ich sehen wollte – ich musste dies umständlich eingeben.

Problem auch praktisch lösbar ist. Dies erscheint zunächst ein offensichtlicher Allgemeinplatz, dem jede:r zustimmen wird, schließlich kann man einen Computer nach drei Jahren offiziell von der Steuer abschreiben und durch ein Gerät ersetzen, das deutlich leistungsfähiger ist und dementsprechend auch deutlich mehr Probleme praktisch lösen kann. Dies erkennt aber einen sehr wichtigen Punkt: Es gibt viele *prinzipiell* leicht lösbare Probleme, die nicht nur jetzt *praktisch* unlösbar sind, sondern *für immer* praktisch unlösbar bleiben werden. Es handelt sich um Probleme, die ich gerne als »in diesem Universum nicht lösbar« bezeichne.

Dazu ein Gedankenexperiment: Wir schreiben auf eine Tafel zehn recht große Zahlen – sagen wir, die Zahlen haben maximal 100 Stellen und sind damit kleiner als ein Googol (die Zahl  $10^{100}$ , nach der Google seinen Namen hat). Zwei Gruppen von Schüler:innen bekommen nun folgende Aufgaben: Die erste Gruppe soll möglichst wenige Zahlen auf der Tafel einkreisen, sodass die Summe *mindestens* ein Googol ist. Die zweite Gruppe soll hingegen möglichst wenige Zahlen einkreisen, sodass die Summe *genau* ein Googol ist. Für beide Gruppen existieren einfache Algorithmen zur Lösung: Die erste Gruppe muss einfach immer so lange die größte noch nicht eingekreiste Zahl auf der Tafel einkreisen, bis die Summe ein Googol überschreitet – bei großen Zahlen ein gewisser Aufwand und man verrechnet sich sicherlich oft, aber im Prinzip in einer Schulstunde machbar. Für die zweite Gruppe ist der Algorithmus auch einfach, wenn auch zeitaufwändiger: Sie kann systematisch alle Möglichkeiten durchprobieren, unterschiedliche Zahlen einzukreisen, und dann die Möglichkeit auswählen, die einerseits die Summe  $10^{100}$  ergibt und andererseits möglichst wenige Zahlen umfasst. Beide Probleme sind also algorithmisch leicht lösbar, wenn auch das zweite aufwändiger erscheint als das erste, und damit ist es auch kein großes Problem, Programme zur Lösung in einer beliebigen gängigen Programmiersprache zu schreiben.

Ganz anders verhält es sich mit der *praktischen* Lösbarkeit der beiden Probleme, denn das zweite Problem – in der Literatur als SUBSETSUM Problem bekannt – ist NP-vollständig und der beste bekannte Algorithmus zur Lösung ist (stark vereinfacht ausgedrückt) genau die gegebene Empfehlung an die zweite Gruppe, »einfach alle Möglichkeiten durchprobieren«.

Nun kann ein moderner Rechner alle Möglichkeiten, unter zehn Zahlen welche auszuwählen und dann die Summe zu bilden, in etwa einer Mikrosekunde durchrechnen. Erhöhen wir die Anzahl der Zahlen auf der Tafel auf 20, so steigt die Zeit auf eine Millisekunde, bei 30 Zahlen ist es eine Sekunde, bei 50 Zahlen bewegen wir uns dann schon im Bereich von Tagen und Wochen, bei 60 Zahlen im Bereich von Jahren, bei 100 Zahlen in Bereichen der Form »so lange, wie das Universum alt ist«.

Hieran wird sich *nichts ändern*, wenn die Computer in Zukunft schneller oder paralleler werden – ja selbst Quantenrechner werden hieran (entgegen der weit verbreiteten Vorstellung) nichts ändern: Baute man einen Quantenrechner, der schlicht jedes Atom im Universum als einen »Prozessor« nutzt, takteten wir ihn mit der höchsten physikalisch vorstellbaren Taktfrequenz (dem Kehrwert der Planck-Zeit von etwa  $5,4 \cdot 10^{-44}$  s, der kleinsten Zeit, in

der ein Informationsaustausch durch Licht möglich ist) und lassen ihn eine Million mal länger rechnen, als das Universum alt ist – der Rechner würde nicht fertig werden, wenn auf der Tafel 500 Zahlen stünden.

Entscheidend ist die Einsicht, dass wir hier und heute auf eine Schultafel Zahlen schreiben können (wenn man klein schreibt, passen 500 darauf), sodass kein Computer in diesem Universum – weder jetzt noch in Zukunft, weder im Keller der NSA noch bei Aliens auf Alpha Centauri – jemals herausfinden wird, welche man einkreisen muss, um als Summe ein Googol zu erhalten.<sup>4</sup>

Diese Einsicht hat aus meiner Sicht viele praktische Auswirkungen: Es sind eben nicht nur die teils etwas esoterischen Probleme der fortgeschrittenen Rekursionstheorie, die von Computern niemals gelöst werden können. Vielmehr werden viele praktische Probleme aus unserem Alltag von Computern auch in Zukunft – während der Lebenszeit der heutigen Schüler:innen, ihrer Kinder und ihrer Kindeskinde – nur graduell etwas besser gelöst werden, niemals aber fundamental besser. Ganz praktisch gesprochen ist zum Beispiel eine (richtig) verschlüsselte E-Mail sicher und *wird dies auch bleiben*. Dies zu wissen, ist wichtig.

## 5 Informatik-Artefakte sind kollaborative Kulturgüter

Der aktuelle Rahmenplan von Berlin und Brandenburg sieht [La15, Teil C, Seite 4] zum Thema »Wechselwirkungen zwischen Informatiksystemen, Mensch und Gesellschaft beurteilen« vor: »Schülerinnen und Schüler erläutern beispielhaft, wie Informatiksysteme den Alltag und die Berufswelt durchdringen und verändern. Dabei diskutieren sie Vor- und Nachteile des Einsatzes von Informatiksystemen.« Dieses Kompetenzziel erscheint mir ebenso sinnvoll wie erstrebenswert, jedoch lässt es einen Aspekt außer acht, der in diesem letzten Abschnitt kurz beleuchtet werden soll und der – so meine Meinung – ebenfalls zu den wichtigsten Eigenschaften der Informatik jetzt und in Zukunft gehört: »Die Gesellschaft« (hier in Form von »Alltag und Berufswelt«) ist nicht nur passive Rezipientin, die von Informatiksystemen durchdrungen wird, es ist umgekehrt auch die Gesellschaft, die Informatiksysteme aktiv erschafft.

Hier lässt sich aus meiner Sicht eine deutliche Änderung in den letzten vielleicht zehn Jahren erkennen in Bezug auf die Frage, *wer* Informatik-Artefakte (sei es Software, Algorithmen, Skripte oder auch Hardware) erschafft. Lange Zeit war die Erstellung von Software, geschweige denn die Herstellung von mikroelektronischen Geräten, nur mit nicht unerheblicher informatischer Vorbildung möglich, benötigte erhebliche technische

---

<sup>4</sup> Disclaimer: Genau genommen liegen die Dinge etwas komplizierter und dem Beispiel unterliegen bestimmte Komplexitätstheoretische Annahmen wie  $NP \neq BPP$ . Es ist aber aus Komplexitätstheoretischer Sicht leicht, andere praktisch relevante (nur etwas weniger anschauliche) Probleme zu finden, die nachweislich und ohne Annahmen auch von Quantenrechnern nur in exponentieller Zeit gelöst werden können.

Expertise und auch technische Ausrüstung. Die Erstellung umfangreicherer Software-Systeme benötigte sorgfältige Planung durch Softwarearchitekt:innen und dies galt sowohl für kommerziell erstellte Software wie auch für freie Software (wo »schlecht geplante« Projekte in einem darwinistischen Prozess beständig durch bessere ersetzt werden).

Die Softwarelandschaft verändert sich seit einiger Zeit massiv und dieser Trend scheint mir fundamental zu sein: Zunächst entsteht überhaupt erstmal eine *Softwarelandschaft* und nicht mehr eine Ansammlung von durch Softwarearchitektur präzise geplanten und von Mauern umgebenen Häusern. Vielmehr wächst eine bunte Vielfalt in Form eines komplexen Ökosystems an Software, in dem viele kleinere – nicht mehr vollständig zu überblickende – Software-Artefakte miteinander vernetzt werden, beispielsweise durch Web-Services oder einfach dadurch, dass sie auf einer gemeinsamen Plattform laufen. Auf der anderen Seite werden immer mehr Menschen auch ganz ohne Informatikvorbildung zu Erzeuger:innen von Informatik-Artefakten beispielsweise in Form von Skripten für Spiele(?)-Plattformen wie Minecraft oder Roblox; durch weitere Fortschritte bei 3D-Druckern wird auch das eigenständige Drucken von Hardware Alltag werden.

Diese Entwicklungen genau vorauszusagen, ist unmöglich und sie ließe sich noch sehr viel intensiver analysieren; jedoch erscheint es mir wichtig, dass Menschen und insbesondere den jetzigen Schüler:innen bewusst ist, dass ihr Alltag und ihre Arbeitswelt weniger von externen Informatik-Artefakten (die ihrer Kontrolle nicht unterliegen) durchdrungen wird, sondern dass sie selbst aktive Akteur:innen bei der Erschaffung, Weiterentwicklung und Steuerung von Informatik-Artefakten sein können und oft sein werden.

## 6 Zusammenfassung

Es gibt eine Reihe von »Basisideen« oder »fundamentalen Einsichten« oder »grundlegenden Beobachtungen«, die eine im wahrsten Sinne des Wortes andere – oder zumindest weitere – Sicht auf die Welt ermöglichen, die originär der Informatik entstammen und die sinnvoll nur im Kontext eines Informatikunterrichts verstanden, erklärt und weiter ausgeführt werden können. Dieser Artikel hat versucht, die (sicherlich kontroverse) These zu untermauern, dass diese Basisideen der Informatik auf einer Stufe stehen sollten in einem schulischen Curriculum wie den Basisideen der »etablierten« Schulfächer wie Physik oder Mathematik, aber auch Kunst oder Deutsch.

Die vorgestellten Ideen entstammten einerseits der Theoretischen Informatik – wodurch sie zumindest eine gewisse Zeitlosigkeit beanspruchen können –, andererseits auch dem Bereich Informatik und Gesellschaft. Sie stellen in der vorgestellten Form zunächst eine Auswahl dar, die als Anstoß für eine weitere Debatte gedacht ist, und die Auswahl könnte und sollte um weitere Ideen ergänzt werden, die ebenfalls sowohl originär der Informatik entstammen, die von fundamentaler Natur sind und die Menschen eine andere Sicht auf ihre Umwelt ermöglichen. Beispieler weiterer solcher Ideen sind die folgenden, die Liste ließe sich zweifelsohne fortsetzen:

- Die Idee der *Modellierung* als Prozess der Abstraktion eines Teils der Wirklichkeit. Sicherlich ist dies zunächst keine reine Idee der Informatik, denn viele Disziplinen arbeiten regelmäßig mit Modellen: Das Spektrum reicht von der Architektur in Form von physischen Modellen über die Physik in Form von die Wirklichkeit approximierenden Theorien bis zur Zoologie in Form von Spezies oder Taxa als Modell einer Gruppe von Lebewesen. Das Spezifische der Informatik scheinen mir Größe und Umfang der Modelle zu sein: Sollen Theorien als Modelle der Wirklichkeit gerade möglichst klein und einfach sein, wollen die Informatik-Modelle einen bestimmten Ausschnitt der Wirklichkeit so umfassend darstellen, dass das Modell im Idealfall identisch wird mit dem Programm.
- Die Idee des *Protokolls* als formalisierte Unterhaltung. Wieder ist die Idee (schon das Wort) alles andere als neu – und wieder steuert die Informatik mit der Idee einer schier grenzenlosen Anzahl sich ständig mittels formaler Protokolle unterhaltender Systeme eine qualitativ neue Sicht auf die Umwelt bei.
- Die Idee der *asymmetrischen Verschlüsselung*. Schon die Idee der Verschlüsselung und ihrer Auswirkungen ist aus meiner Sicht von fundamentaler Natur – die Idee von Diffie und Hellman [DH76] und ihre Umsetzung durch Rivest, Shamir und Adleman [RSA78] der asymmetrischen Verschlüsselung aber ist ebenso kontraintuitiv wie fundamental und für den Alltag relevant.

Wenn man dem Argument folgt, dass die in der Arbeit vorgestellten Ideen eine neue, reichere Sicht auf »die Realität« ermöglichen, so folgt daraus aus meiner Sicht auch die Notwendigkeit, diese Erkenntnisse in ähnlicher Breite und Art bereits in der Schule zu unterrichten, wie dort von Atomen, Bakterien oder der Schwerkraft berichtet wird.

## Literatur

- [Bo54] Boole, G.: An Investigation of the Laws of Thought, on Which are Founded the Mathematical Theories of Logic and Probabilities. MacMillan und Co., Cambridge, 1854.
- [Br04] Brinda, T.; Claus, V.; Diethelm, I.; Humbert, L.; Johlen, D.; Magenheimer, J.; Micheuz, P.; Modrow, E.; Puhlmann, H.; Scheel, O.; Schneider, M.; Schubert, S.; Schulte, C.; Schwill, A.; Zündorf, A.: Zweite Dagstuhler Empfehlung zur Aufnahme des Fachs Informatik in den Pflichtbereich der Sekundarstufe I, <http://www.informatikdidaktik.de/hyfisch/informieren/politik/dagstuhlerempfehlung2004.htm>, Zugriff 2021-07, 2004.
- [Br76] Brauer, W.; Claus, V.; Deussen, P.; Eickel, J.; Haacke, W.; Hosseus, W.; Koster, C.; Ollesky, D.; Weinhart, K.: Zielsetzungen und Inhalte des Informatikunterrichts. Zentralblatt für Didaktik der Mathematik 1/, 1976.

- [Bu92] Buse, D.; Claus, V.; Graf, K.-D.; Hahlweg, E.; Koerber, B.; Loos, R.; Oberschelp, W.; Schubert, S.; Schwill, A.; Stimm, H.: Dagstuhler Empfehlung zur Aufnahme des Fachs Informatik in den Pflichtbereich der Sekundarstufe II, <http://www.informatikdidaktik.de/HyFISCH/Informieren/politik/DagstuhlerEmpfehlung1992.htm>, Zugriff 2021-07, 1992.
- [Ca35] Capelle, W.: Die Vorsokratiker, Fragmente und Quellenberichte. Kröner, Leipzig, 1935.
- [DH76] Diffie, W.; Hellman, M. E.: New Directions in Cryptography. IEEE Transactions on Information Theory 22/6, S. 644–654, 1976.
- [Ge15] Gemulla, R.; Martens, W.; Schöning, J.; Schulte, C.: Dagstuhl Manifesto. Informatik-Spektrum 38/, Weitere Autoren sind die Teilnehmerinnen und Teilnehmer des Seminars. Auch veröffentlicht als »3. Dagstuhl-Erklärung zur Informatischen Bildung in der Schule 2015 der Gesellschaft für Informatik«, S. 243–252, 2015.
- [Ge90] Gerhardt, C. J., Hrsg.: Die philosophischen Schriften von Gottfried Wilhelm Leibniz. Weidmann, 1890.
- [GH14] Großmann, S.; Hertel, I., Hrsg.: Zur fachlichen und fachdidaktischen Ausbildung für das Lehramt Physik. Deutsche Physikalische Gesellschaft, 2014.
- [Gö31] Gödel, K.: Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I. Monatshefte für Mathematik und Physik 38/, S. 173–198, 1931.
- [Ko05] Konferenz der Kultusminister der Länder: Beschlüsse der Kultusministerkonferenz, Bildungsstandards im Fach Physik für den Mittleren Schulabschluss. Beschluss vom 16.12.2004, Luchterhand, 2005.
- [La15] Landesinstitut für Schule und Medien Berlin-Brandenburg: Rahmenlehrplan Jahrgangstufen 1–10. Senatsverwaltung für Bildung, Jugend und Wissenschaft Berlin; Ministerium für Bildung, Jugend und Sport des Landes Brandenburg, 2015.
- [Me43] Menabrea, L. F.: Sketch of the Analytical Engine Invented by Charles Babbage. Scientific Memoirs 3/, Entscheidend sind hier die »Notes upon the memoir by the translator«, welche Ada Lovelace war, 1843.
- [OR23] Ogden, C. K.; Richards, I. A.: The Meaning of Meaning. Kegan Paul, London, 1923.
- [Po46] Post, E. L.: A Variant of a Recursively Unsolvable Problem. Bulletin of the American Mathematical Society 52/, S. 264–268, 1946.
- [Re16] Reischuk, R., Hrsg.: Creative Mathematical Sciences Communication Conference (CMCS 2016), Institut für Theoretische Informatik, Universität zu Lübeck, Lübeck, Okt. 2016.

- [Ri20] Richard Schwarz Lutz Hellmig, S. F.: Informatikunterricht in Deutschland – Eine Übersicht, Zusammenfassung von [Sc20], [https://pidi.informatik.uni-rostock.de/storages/uni-rostock/Alle\\_IEF/Inf\\_PI/files/Vergleich\\_IU\\_2020\\_2020-11-23.pdf](https://pidi.informatik.uni-rostock.de/storages/uni-rostock/Alle_IEF/Inf_PI/files/Vergleich_IU_2020_2020-11-23.pdf), Zugriff 2021-07, 2020.
- [RSA78] Rivest, R. L.; Shamir, A.; Adleman, L.: A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM* 21/2, S. 120–126, Feb. 1978, ISSN: 0001-0782, URL: <https://doi.org/10.1145/359340.359342>.
- [Sc20] Schwarz, R.: Informatikunterricht in den Bundesländern Deutschlands im Jahr 2020, Hausarbeit im Rahmen der Ersten Staatprüfung für das Lehramt an Gymnasien, Universität Rostock, 2020.
- [Sc94] Schwill, A.: Fundamental Ideas of Computer Science. *Bull. European Assoc. for Theoretical Computer Science* 53/, 1994.
- [Tu37] Turing, A. M.: On Computable Numbers, with an Application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society* s2-42/1, S. 230–265, Jan. 1937, ISSN: 0024-6115, eprint: <https://academic.oup.com/plms/article-pdf/s2-42/1/230/4317544/s2-42-1-230.pdf>, URL: <https://doi.org/10.1112/plms/s2-42.1.230>.