Vorschlag zur Ausgestaltung der Algorithmik im gymnasialen Mathematikunterricht

Dr. Martin Guggisberg
martin.guggisberg@unibas.ch

Christian Datzko
christian@datzko.ch

27. November 2012

Zusammenfassung

Die Autoren machen einen Vorschlag, wie die in der aktuellen Planung vorgesehene Stunde „Algorithmik“ im Mathematikunterricht so mit Inhalten gefüllt werden kann, dass sie aus Sicht der Mathematik eine sinnvolle Ergänzung ist, dass sie gleichzeitig den Ansprüchen und Wünschen moderner Mathematik als auch den bewährten Grundlagen traditioneller Mathematik genügt, und dass sie Maturandinnen und Maturanden eine hinreichende Grundlage der Allgemeinbildung bietet. Hierfür werden didaktische Abwägungen gemacht, klassische Beispiele gebracht und ein konkreter Lehrplanvorschlag unterbreitet.

Inhaltsverzeichnis

1 Motivation

Im Kanton Basel-Stadt werden die Weichen für eine Neuordnung der Gymnasien gestellt. Es werden Lehrpläne neu entworfen und dabei Althergebrachtes in Frage gestellt und Neues vorgeschlagen. Dabei ist in der Stundentafel für das Grundlagenfach Mathematik, die im Erziehungsrat von Basel-Stadt am 13. Juni 2012 beschlossen wurde, vorgesehen, dass eine Jahresstunde für „Algorithmik“ verwendet wird[1]. Die Ausgestaltung der kantonalen Lehrpläne wird momentan auf Ebene der Fachkonferenzen der Gymnasien sowie auf Ebene der Zentralen Fachkonferenzen diskutiert. An diese wenden sich die Autoren mit diesem Dokument im Sinne eines Vorschlags für die Ausgestaltung der Inhalte.

Wir machen diesen Vorschlag, da wir der Überzeugung sind, dass eine fundierte Umsetzung der Grundideen zur Algorithmik für Maturandinnen und Maturanden eine weitere wertvolle Grundlage für das Studium vieler angewandter Fächer gibt. Neben Theorie, Beobachtung und Experiment ist die Simulation, die heutzutage fast ausschliesslich auf Algorithmen basieren, eine weit verbreitete wissenschaftliche Methode geworden, so dass Maturandinnen und Maturanden im Sinne einer allgemeinen Studierfähigkeit auch auf dieses insofern vorbereitet sein müssen, als dass sie die grundlegenden Konzepte verstanden haben. Gleichzeitig erleben Maturandinnen und Maturanden in der Welt ein Füllhorn von Anwendungen der vorgeschlagenen Inhalte, so dass eine über die Studierfähigkeit hinausgehende Praxisrelevanz ebenfalls gegeben ist.

Dieses Dokument spricht bewusst keine ICT-Fähigkeiten an, da diese zum einen wenig mit dem Gegenstand des Algorithmus zu tun haben und zum anderen laut derzeitiger Planung möglicherweise sowieso an anderer Stelle Eingang in den Lehrplan finden werden. Das soll aber nicht heissen, dass nicht einzelne Elemente dieses Vorschlags auch zum Beispiel mit Hilfe von Tabellenkalkulationen behandelt werden können.

2 Allgemeine didaktische Überlegungen

2.1 Begriff des Algorithmus

Nach Schubert und Schwill (2004, S. 5) ist ein Algorithmus ein „mit formalen Mitteln beschreibbares, mechanisch nachvollziehbares Verfahren zur Lösung einer Klasse von Problemen“. Charakteristische Eigenschaften eines Algorithmus sind nach Strauch und Rehm (2007, S. 5):

- Diskretheit – Ein Algorithmus besteht aus einer Folge von Schritten.
- Determiniertheit – Bei gleichen Startbedingungen erzeugt er stets dasselbe Endergebnis.
- Eindeutigkeit – Nach jedem Schritt lässt er sich auf höchstens eine Art fortsetzen.
- Endlichkeit - Er endet nach endlich vielen Schritten.

Geometrie, Algebra und Analysis befassen sich mit Form und Struktur mathematischer Objekte häufig in einer statischen Sichtweise. Ein Algorithmus jedoch beschreibt in vielen Fällen

[1] „Das Thema Algorithmen wird in den Lehrplan des Fachs Mathematik integriert (Abteilungsunterricht).“ (Arbeitsgruppe Porträt/Stundentafel/Lehrplanarbeit des Stufenprojekts Gymnasien, 2012, S. 3)

ein Verfahren, durch welches sich mathematische Objekte im Verlauf der Zeit verändern. Eine dynamische Sichtweise beim Arbeiten mit Algorithmen kann die mathematische Denk- und Arbeitsweise von Schülerinnen und Schülern erweitern.

So ist zum Beispiel die Suche nach einer Ableitungsfunktion häufig als statische Fragestellung formuliert, als Schülerantwort wird ein neue Funktion eingefordert. Die Aspekte von momentanen Änderungsraten, welche bei physikalischen Anwendungen im Vordergrund stehen, werden im Mathematikunterricht selten angesprochen. Auch arithmetische und geometrische Folgen, die dynamische Prozesse beschreiben können, werden bevorzugt mit Hilfe einer geschlossenen Formel für a_n bearbeitet. Selbst eine typischerweise rekursiv definierte Folge wie zum Beispiel die Fibonacci-Folge (siehe S. 7) wird im Unterricht häufig mit Hilfe einer invarianten Rekursionsformel beschrieben.

Dem gegenüber steht der Algorithmus, der in sich dynamisch ist. Die Möglichkeit, Probleme schrittweise approximativ anzugehen oder für nicht lösbare oder zumindest praktisch nicht lösbare Probleme durch Näherungsverfahren numerische Lösungen zu finden, zeigt das enorme Potenzial numerischer Lösungsverfahren. Die oben geforderte mechanisch Nachvollziehbarkeit eines Algorithmus bedeutet zudem, dass er per se automatisierbar ist, also von Maschinen ausgeführt werden kann. Das ermöglicht, von Kalkülen zu konkreten, angewandten Problemstellungen überzugehen (Leuders, 2003, S. 199).

Die Ausweitung der Mathematik auf dynamische Aspekte ist nicht neu. Alonzo Church, Noam Chomsky, Andrei Markow, Charles Babbage und Alain Turing waren alles Mathematiker, die die Methoden der Mathematik wesentlich weiterentwickelten, indem sie sich mit Verfahren befassten, die der oben genannten Definition eines Algorithmus entsprechen. Weitere historische Beispiele finden sich im Kapitel 3.1.

Auch in der aktuellen Mathematikforschung wird vermehrt mit Hilfe von Algorithmen gearbeitet. Die Experimentelle Mathematik zum Beispiel ist ein sehr junges Forschungsgebiet der Mathematik, welches erst in den letzen drei Dekaden entstanden ist. Numerische Mathematik und Computeralgebrasysteme werden als Werkzeuge zur Erforschung von mathematischen Problemen verwendet. Prominente Beispiele sind das Vierfarbenproblem oder die Keplersche Vermutung. Bei Borwein et al. (2010) werden weitere Erfolge dieses jungen Forschungszweigs aufgezeigt.

2.2 Braucht Mathematikunterricht Algorithmik?

Mathematikdidaktik ordnet sich in der Regel dem Zeitgeist unter. Wie Tietze et al. (2000a, S. 43 ff.) ausführen, war zum Beispiel in den Achzigerjahren der Rechnereinsatz für numerische Methoden im Mathematikunterricht „in“, seit Anfang/Mitte der Neunzigerjahre wird eher Anwendungssoftware wie Dynamische Geometriesoftware oder Computeralgebrasysteme verwendet. Durch das Aufkommen von bezahlbaren Taschenrechnern mit Computeralgebrasystem-Funktionalität wurden diese immer häufiger im Mathematikunterricht verwendet. Momentan kann aber auch ein gegensätzlicher Trend beobachtet werden, da der darauf folgende Verlust von manuellen Fertigkeiten zum Beispiel beim Lösen von Gleichungen nicht immer als irrelevant betrachtet wird. Bei einer mathematikdidaktischen Abwägung darf also der Algorithmus nicht zum Selbstzweck erhoben werden sondern muss aus der Notwendigkeit der Mathematik heraus begründet werden.

Tietze et al. (2000a, S. 13) schreiben in Bezug auf Klafkis neuen Allgemeinbildungsbegriff (Klafki, 1994): „Mathematik beinhaltet dann nicht in erster Linie das formale Herleiten mathematischer Aussagen, sondern das experimentelle Arbeiten mit konkreten Objekten und Modellen, das experimentelle Untersuchen mathematischer Sachverhalte mit dem Computer und dem Taschenrechner, die Auseinandersetzung mit Problemen der Realität mit Hilfe von Mathematik in vielfältiger Form (wie Graph, Formel, Algorithmus, Programm) usw." sowie auf S. 38: „*Algorithmus* (als mechanische Prozedur für das Berechnen oder das Fällen von Entscheidungen, als mathematischer Kalkül, als Rechner-Programm), *Funktion* (funktionale Abhängigkeit, eindeutige Zuordnung, Abbildung, Transformation, Operator), *Approximation* und *Modellbildung* werden allgemein als grundlegende Ideen der Mathematik akzeptiert." Oldenburg (2011, S. 1) bezieht sich ebenfalls darauf, wenn er schreibt: „Das bedeutet, dass die Idee, mathematische Kenntnisse in Form von Verfahren zu kondensieren, mit denen man in der Lage ist, eine Vielzahl von Fragestellungen systematisch zu bearbeiten, zum Wesen der Mathematik gehört".

Demnach ist die Beschäftigung mit Algorithmen ein wesentlicher Teil von Schulmathematik.

Schwank (1993) zitiert nach Sjuts (1999, S. 62) hat hierzu grundlegende lernpsychologische Untersuchungen zu Lern- und Denkstrukturen von Schülerinnen und Schülern durchgeführt und festgestellt: „‚Bei einer Präferenz für eine prädikative kognitive Struktur liegt eine Präferenz vor, in einer gegebenen Situation die statische Beziehungshaftigkeit auszudrücken, das Augenmerk auf die Strukturierung und ihre Beschreibung zu lenken. Bei einer Beschreibung von Handlungen steht die Darstellung von Beziehungen zwischen Einsatz und Ergebnis im Vordergrund. Die Sensibilisierung für die Genauigkeit in diesem Bereich ist erheblich größer als die Fähigkeit, komplexe Prozesse wahrzunehmen und zu analysieren.' Das heißt, ‚daß für den Umgang mit solchen Beziehungen und Strukturen eine größere Erfahrung und Sicherheit und daraus resultierend eine größere Schnelligkeit und geringere Fehleranfälligkeit sich ergeben. Bei einer Vorliebe für eine funktionale kognitive Struktur stehen nur wenige Beziehungen und Strukturanalysen unmittelbar zur Verfügung, dafür gibt es aber einen ausgeprägtes Sensorium für Prozesse und das Denken in Wirkungsweisen. Dazu gehört auch eine unmittelbare (d. h. nicht als Ergebnis punktueller Berechnungen) vorhandene Einsicht in die Kategorien von Mühe und Aufwand von Prozessen. Da bei einer Vorliebe für eine solche kognitive Struktur die Beziehungen und Beschreibungen nicht grundlegend sind, ist auch das Interesse, solche Beschreibungen genau und präzise vorzunehmen, wenig ausgeprägt.'" Demnach ist für den Teil der Schülerinnen und Schüler, die funktional denken, eine einseitige Festlegung auf eine statische Sichtweise der Mathematik erschwerend. Eine Aufweitung der Sicht auf dynamische Prozesse könnte solchen Schülerinnen und Schülern helfen, einen besseren Zugang zur Mathematik im Allgemeinen zu erhalten.

Gleichzeitig engt man die Sicht für prädikativ denkende Schülerinnen und Schüler aber nicht ein. Sjuts (1999, S. 128) schreibt hierzu im Hinblick auf Cohors-Fresenborg und Kaune (1993): "Weil sich der Algorithmusbegriff auffassen läßt als die Mathematisierung des Handlungsbegriffs, es beim letzteren eine prädikative und eine funktionale Akzentuierung gibt, je nachdem, ob mehr die Ergebnisse oder die Prozesse ihres Zustandekommens im Vordergrund stehen, muß der Mathematikunterricht bei der Wahl der algorithmischen Beschreibungsrepräsentation beide Auffassungen berücksichtigen".

Die Beschäftigung mit Algorithmen kann also eine Chance für Schülerinnen und Schüler bieten, deren kognitive Strukturen im Mathematikunterricht bisher nicht hinreichend berücksichtigt wurden ohne einen Nachteil für Schülerinnen und Schüler zu bieten, die bisher bevorzugt angesprochen wurden.

Dieser hier vorliegende Vorschlag soll keine versteckte Einführung eines Grundlagenfachs Informatik sein. Rechner oder Computer sollen als Werkzeuge für das Lösen konkreter mathematischer Fragestellung rund um die Thematik Algorithmik eingesetzt werden können. Für eine umfassende und weiterführende Einführung in die Informatik sehen wir das Ergänzungsfach Informatik. Aktuelle Artikel zu Bildungsstandards in der Sekundarstufe I (Arbeitskreis Bildungsstandards der Gesellschaft für Informatik e.V., 2008) und in der Sekundarstufe II (Baumann, 2011) zeigen, dass ein Grundlagenfach Informatik deutlich über die in diesem Vorschlag genannten Inhalte hinaus gehen muss.

2.3 Neue Chancen für den Mathematikunterricht

Über das Ergebnis hinaus, dass der Algorithmus eigentlich einen sinnvollen Platz im Mathematikunterricht hat, bietet die Algorithmik neue Chancen. So können mit Hilfe von Automatisierung mathematische Ideen und Sachverhalte experimentell erforscht werden. Tietze et al. (2000a, S. 46) schreiben hierzu[2]: „Ganz neue Perspektiven für den Mathematikunterricht beinhaltet der Rechner als *Entdecker*. Er ermöglicht ein quasi experimentelles Umgehen mit mathematischen Gegenständen, indem man durch systematisches Variieren eines mathematischen Sachverhalts versucht, zu Hypothesen zu gelangen. Das gilt insbesondere für die Betrachtung von Funktions- und Kurvenscharen, für die Untersuchung von Zahlenfolgen, aber auch für die Untersuchung von Matrizenfolgen, etwa bei Markoff-Ketten, aber auch im Zusammenhang mit dem Modellbilden“.

Konkret stellen sich Tietze et al. (2000a, S. 211) zum Beispiel die Approximation durch Iteration und Rekursion als „besondere Form der universellen Idee ‚Algorithmus‘“ vor, z. B. durch Approximationsverfahren von irrationalen Zahlen (z. B. durch Intervallschachtelung, das Heronverfahren oder einer Annäherung von π durch Polygone), durch Approximationsverfahren von Nullstellen, Lösen von Gleichungen oder Annähern von Funktionen und Ableitungen.

Gleichzeitig ist die Beschäftigung mit dem Entwerfen von Algorithmen auch eine Bündelung und damit Sicherung verschiedener mathematischer Kompetenzen. Dabei wird angewendet (vergleiche Ständige Konferenz der Kultusminister der Länder in der Bundesrepublik Deutschland (2012)), ...

- mathematische Probleme zu lösen.
- mathematische Darstellungen zu verwenden (viele mathematische Konzepte lassen sich auf Algorithmenstrukturen abbilden).
- mathematisch zu argumentieren (die Korrektheit (oder auch Unkorrektheit) eines Algorithmus zur mathematischen Berechnung muss immer wieder (sich selbst, Mitschülern, der Lehrkraft) erklärt werden).
- mathematisch zu modellieren (es werden viele inner- wie aussermathematische Situationen mit den Mitteln des Algorithmus beschrieben (modelliert) und die Ergebnisse werden wieder interpretiert).
- mit symbolischen, formalen und technischen Elementen der Mathematik umzugehen.
- mit Menschen zu kommunizieren, auch wenn Algorithmen primär der Kommunikation mit einer Maschine dienen.

[2] Der Begriff „Rechner“ wird im Kontext von Tietze et al. (2000a) nicht als Taschenrechner sondern allgemein als berechnende Maschine, explizit auch als Computer, verstanden.

3 Klassische Beispiele für Algorithmen in der Mathematik

Um den abstrakten Begriff des Algorithmus in der Mathematik zu konkretisieren, sollen hier einige Beispiele repräsentativ für die mögliche Umsetzung im Unterricht stehen. Diese Sammlung soll aber keine Forderung für die Umsetzung im Unterricht darstellen sondern aufzeigen, wie vielfältig die Anwendungen in der Mathematik sind. Weitere Beispiele für Algorithmen in der Mathematik können auch bei Oldenburg (2011) und Gressley Freimann und Guggisberg (2011) gefunden werden.

3.1 Historische Algorithmen

3.1.1 Heron-Verfahren und Intervallschachtelungs-Verfahren

Das Heron-Verfahren (als solches mindestens 2100 Jahre alt, vermutlich jedoch schon über 3750 Jahre alt) ist ein iteratives Näherungsverfahren zum numerischen Berechnen von Wurzeln. Ausgehend von einer geometrischen Veranschaulichung wird ein Rechteck mit der Fläche des Wurzelradikanten einem Quadrat mit gleicher Fläche angenähert.

Das Intervallschachtelungs-Verfahren dient demselben Zweck, ist etwas leichter verständlich, konvergiert jedoch langsamer.

3.1.2 Euklidischer Algorithmus

Der euklidische Algorithmus (mindestens 2300 Jahre alt) wird häufig als der erste nichttriviale Algorithmus der Welt bezeichnet. In seiner ursprünglichen Form wird durch wiederholte Subtraktion der grösste gemeinsame Teiler zweier natürlicher Zahlen berechnet. Er kann sowohl iterativ als auch rekursiv formuliert werden.

3.1.3 Newton-Verfahren

Das Newton-Verfahren (ca. 340 Jahre alt) ist ein iteratives Näherungsverfahren zum numerischen Berechnen von Lösungen von Gleichungen beziehungsweise Gleichungssystemen. Klassisch wird es zum Bestimmen von Nullstellen von Funktionen verwendet.

3.1.4 Horner-Schema

Das Horner-Schema (mindestens 200 Jahre alt, wahrscheinlich jedoch schon über 700 Jahre alt) ist eine weitere Möglichkeit, Nullstellen von Polynomen zu berechnen, auch in Verbindung mit dem Newton-Verfahren.

3.1.5 Gaußsches Eliminationsverfahren

Das Gaußsche Eliminationsverfahren (von Gauß vor ca. 200 Jahren beschrieben, in Ansätzen jedoch schon ca. 2000 Jahre alt) ist ein universelles Verfahren zur Lösung linearer Gleichungssysteme mit n Gleichungen und n Variablen. Es besteht im Kern darin, ein Gleichungssystem so umzuformen, dass zunächst die Gleichungen so umgeformt werden, dass eine Treppenform entsteht, so dass bei der letzten Gleichung die Lösung für eine Variable direkt ausgerechnet werden kann. Durch sukzessives Rückeinsetzen kann so die Lösung für alle Variablen direkt berechnet werden.

Das Gaußsche Eliminationsverfahren sowie ähnliche Verfahren werden für sehr viele Simulationen in den verschiedensten Wissenschaften verwendet. Auch Tietze et al. (2000b, S. 71) verweisen darauf.

3.1.6 Numerische Differentiation und Numerische Integration

Numerische Differentiation ist eine direkte Anwendung des Differenzenquotienten in Fällen, wo eine symbolische Differentiation nicht möglich oder umständlich ist. Sie kann auch zur Ideenfindung für das Ergebnis einer symbolischen Differentiation verwendet werden, wenn die symbolische Differentiation noch nicht bekannt ist.

Numerische Integration ist die analoge Umsetzung der Numerischen Differentiation für die Integration. Die Anwendungen sind ebenfalls das Finden von Integralen bei nicht-existenten oder umständlichen symbolischen Integralen.

3.1.7 Matrizenrechnung

Nach Tietze et al. (2000b, S. 71) sind Matrizenkalkül sowie Koordinaten- und Matrizentransformation klassische Beispiele für eine Algorithmisierung. Aufgrund der grossen Menge von simplen Berechnungen, die für Matrizenberechnungen notwendig sind, ist hier eine Auslagerung der Berechnungen tatsächlich sinnvoll.

3.1.8 Folgen und Reihen

Für eine phänomenale Betrachtung von Folgen und Reihen, die nicht gerade arithmetisch oder geometrisch sind, kann eine algorithmische Betrachtung ebenfalls helfen[3].

Die Fakultät ist eine Folge, die entweder als $n! = \prod_{i=1}^{n} i$ oder rekursiv als $a_1 = 1$, $a_n = n \cdot a_{n-1}$ definiert werden kann. An ihr kann das Prozessartige von Folgen gut verdeutlicht werden.

Die Fibonacci-Folge zum Beispiel ($a_1 = a_2 = 1$, $a_n = a_{n-1} + a_{n-2}$) beschreibt die Entwicklung von verschiedenen Wachstumsprozessen. Sie ist eine klassische Modellierung von Prozessen.

Das Collatz-Problem ist eines der grossen ungelösten mathematischen Probleme. Es ist eine Menge von Folgen, die dem folgenden Prinzip folgen:

$$a_1 \in \mathbb{N}, a_n = \begin{cases} \frac{a_{n-1}}{2} & \text{wenn } a_n \text{ gerade ist} \\ 3 \cdot a_{n-1} + 1 & \text{sonst} \end{cases}$$

Die Collatz-Vermutung ist, dass die Folgen für alle natürlichen Zahlen als Startwert irgendwann in dem Zyklus $(4, 2, 1)$ enden.

[3] Es sei an dieser Stelle erwähnt, dass das alleinige Bearbeiten von Folgen und Reihen noch kein hinreichendes Bearbeiten von Algorithmen bedeutet, da relevante Aspekte von Algorithmen fehlen, wie an der Definition eines Algorithmus im Kapitel 2.1 erläutert und im Kapitel 4.1 ausgeführt.

3.2 Moderne Entwicklungen

3.2.1 Diskrete Optimierung

Wie finde ich den kürzesten Weg, wie verläuft die beste Rundreise oder wie kann ein Rucksack optimal gepackt werden? Bei diesen Fragestellungen handelt es sich um diskrete Optimierungsprobleme. Die Lösung dieser Problemklasse wird schwierig bei grossen Problemgrössen. Für kleine Problemgrössen jedoch können mit Hilfe von Graphen Lösungsstrategien algorithmisch mit Bleistift und Papier erläutert werden.

Letztlich sind viele der Optimierungsprobleme numerische Probleme, die eine minimale oder maximale Teilmenge unter bestimmten Einschränkungen auswählen.

3.2.2 L-Systeme

Mit Hilfe von L-Systemen kann die biologische Entwicklung von Pflanzen durch einen mathematischen Formalismus beschreiben werden. Das wesentliche Prinzip dieser Beschreibung besteht in der sukzessiven Ersetzung von Einzelteilen eines einfachen Objektes mittels festgelegter Regeln. In den Fachbüchern von Rozenberg und Salomaa (1980) und Prusinkiewicz und Lindenmayer (1990) werden zahlreiche Anwendungen von L-Systemen aufgezeigt. Im Internet lassen sich viele interaktive Applets zu unterschiedlichen Simulationen von Pflanzenwachstum finden.

L-Systeme sind letztlich Anwendungen von simplen Funktionen mit teilweise erstaunlich komplexen Auswirkungen.

3.2.3 Generativ erzeugte Strukturen

In der zeitgenössischen Architektur werden häufig generativ erzeugte Muster und Formen eingesetzt. Terzidis (2009) zeigt, wie durch Algorithmen komplexe visuelle Muster und Strukturen erzeugt werden können. Bohnacker et al. (2012) beschreiben in ihrem Buch zahlreiche Designs von visuelle Muster und Strukturen, welche durch unterschiedliche Algorithmen erzeugt werden können.

Generativ erzeugte Muster sprechen Konzepte an, die über simple Kongruenz oder Symmetrie hinaus gehen, aber dennoch immer wieder gerne als mathematische Erzeugnisse präsentiert werden, da sie von vielen Menschen als schön empfunden werden und gleichzeitig viel Mathematik in ihnen steckt.

3.2.4 Scriptfähige Dynamische Geometriesoftware

Programme wie EUKLID DynaGeo, das kostenlos verfügbare GeoGebra oder entsprechende Taschenrechneranwendungen haben als Anwendungsprogramme zum Zeichnen von geometrischen Objekten oder von Funktionen eine Menge interessanter Ansätze gebracht. Durch Schieberegler kann man zudem gut Invarianten oder Zusammenhänge aufzeigen, sogar in Form von Ortskurven.

Die neueste Entwicklung in diesen Programmen ist, dass ihnen eine vollwertige Scriptsprache an die Seite gestellt wird, die dynamische Veränderungen vollständig modellieren können.

Somit können ohne grossen technischen Aufwand dynamische Prozesse visualisiert und deren funktionale Abläufe deutlich gemacht werden.

3.3 Grenzüberschreitungen zur anderen Fächern

3.3.1 Kompression

Bei Kompressionsverfahren werden unwichtige oder eher unwichtigere Informationen weggelassen, um Speicherplatz oder Zeit zu sparen. Klassisch wird dieses Prinzip in der Informatik bei Bildern angewendet. So kann man zum Beispiel ein Bildes Helligkeit, Sättigung und Farbton aufteilen. Wenn man dann zum Beispiel die Auflösung der Sättigung auf 25% reduziert, hat man Speicherplatz gespart, nach Rekombination jedoch fast dasselbe Bild wie vorher, wie das folgende Beispiel deutlich macht.

Original, Rekombination, Helligkeit, Sättigung ($\frac{1}{4}$ der Auflösung), Farbton

Dieses Beispiel verwendet Erkenntnisse des Bildnerischen Gestaltens, nutzt mathematische Modelle zum Umrechnen der Farbmodelle und wird für Fotos und Filme (mit fortgeschritteneren Algorithmen) tagtäglich angewendet.

3.3.2 Populationsdynamik

Wachstumsmodelle die über die Exponentialfunktion hinausgehen lassen sich häufig nicht mehr einfach mit Hilfe einer geschlossenen Formel beschreiben. Daher bieten sich hier ebenfalls Modellierungen mit Hilfe von numerischen Annäherungen an. So ist zum Beispiel die Differentialgleichung für logistische Wachstum $f'(x) = k \cdot f(x) \cdot (G - f(x))$ einfacher verständlich als die geschlossene Formel $f(x) = G \cdot \frac{1}{1+e^{-k\cdot G\cdot x}\cdot\left(\frac{G}{f(0)}-1\right)}$.

Dieses Beispiel lässt sich auf viele Beispiele aus der Biologie oder Geographie anwenden und ist nicht zuletzt aufgrund der Bevölkerungsentwicklung auf der Erde immer wieder ein aktuelles

Thema. Mit Hilfe von Algorithmen können mathematische Modelle einfach ausprobiert und bewertet werden.

3.3.3 Kryptographie

Neben der Faszination, Nachrichten zu verschlüsseln und unbekannte Codes zu knacken ist die Kryptographie eine der klassischen Anwendungen der Mathematik. Neben den klassischen kryptographischen Verfahren wie Rotationschiffren, monoalphabetischen und polyalphabetischen Verschlüsselungen sowie Autokey-Verschlüsselungen spielen moderne kryptographische Verfahren wie die Public-Key-Kryptograhpie eine immer wichtigere Rolle im alltäglichen Leben. So kann man zum Beispiel das RSA-Verfahren, das auf modularem Potenzieren als Falltürfunktion beruht, mit einfachen Zahlen auch mit Schülerinnen und Schülern bearbeiten.

Dieses Beispiel verwendet mathematische Methoden wie die Modulo-Funktion sowie Falltürfunktionen. Gleichzeitig werden Fragen der Codierung von Informationen sowie Abschätzungen zu Komplexitäten von Algorithmen verwendet. Moderne kryptographische Verfahren werden tagtäglich in Internet-Browsern, bei Bankomaten oder beim Telefonieren mit dem Natel verwendet.

3.3.4 Suchen & Sortieren

Die Frage nach effizientem Suchen und effizientem Sortieren ist zunächst eine ureigene Frage der Informatik, deren Beantwortung jedoch viele mathematische Hilfsmittel verwendet. Vor allem die approximative Laufzeit- und Speicherbedarfsanalyse eines Algorithmus ist hierfür relevant. So braucht der naive Ansatz des Sortierens durch Auswählen oder Einfügen in der Regel quadratisch so viel Zeit wie die zu sortierende Menge gross ist im Vergleich zu intelligentem Sortieren durch Quicksort, das bei einer sortierenden Menge von n Elementen nur $\mathcal{O}(n \cdot \log n)$ Zeit braucht.

Dieses Beispiel ist ein klassisches Beispiel dafür, dass auch in der Informatik Methoden der Mathematik dringend gebraucht werden.

4 Vorschlag für einen Lehrplan-Teil Algorithmik

Die Idee, Algorithmik in einen Mathematiklehrplan zu integrieren, ist in der Schweiz nicht neu. Seit 2010 gilt an der Kantonsschule Zug ein Lehrplan für das kantonale Zusatzfach Informatik (Fachschaft Informatik der Kantonsschule Zug, 2010). Dieser besteht aus einem ICT-Teil und einem Algorithmik-Teil, wobei der Algorithmik-Teil einen zeitproportionalen Anteil an der Mathematiknote ausmacht, also faktisch Teil des Mathematikunterrichts ist (Fachschaft Informatik der Kantonsschule Zug, 2010, S. 6). Dieser hier vorliegende Vorschlag für einen Lehrplan-Teil Algorithmik basiert in weiten Teilen auf diesem Lehrplan, allerdings in enger Abgleichung mit den oben ausgeführten allgemeinen didaktischen Überlegungen (Kapitel 2) und den klassischen Beispielen (Kapitel 3).

4.1 Inhalte

Im Zentrum des Algorithmik-Unterrichts sollten die *Problemanalyse* als Methode zum systematischen Herangehen an komplexe Fragestellungen, *Verfahren zum strukturierten Lösen von Problemen* als Werkzeugkasten mit klasssischen Methoden, *Handlungsabläufe* als formelle Darstellung von Lösungswegen sowie die *Informationsrepräsentation in digitalen Systemen* als Wissensgrundlage für konkrete Umsetzungen behandelt werden. Diese Inhalte werden jeweils in Form von zu erreichenden Kompetenzen formuliert.

4.1.1 Problemeanalyse

Die Schülerinnen und Schüler können...

- Probleme systematisch angehen.
- Wichtiges von Unwichtigem unterscheiden.
- ein Problem in Teilprobleme aufteilen.
- wiederverwendbare Lösungsverfahren entwickeln und auf wiederholt auftretenden Teilprobleme anwenden können.
- Grenzen der Lösbarkeit von Problemen benennen.

4.1.2 Verfahren zum strukturierten Lösen von Problemen

Die Schülerinnen und Schüler können...

- Verfahren zur strukturierten Darstellung (z. B. Flussdiagramme, Struktogramme, ...) von Problemlösungen verwenden.
- iterative und rekursive Lösungen von Problemen finden.

4.1.3 Handlungsabläufe

Die Schülerinnen und Schüler können...

- Beispiele für Handlungsabläufe im Alltag und in der Mathematik benennen.
- die grundlegenden Bausteine von Handlungsabläufen anwenden: Zuweisung von Variablen, schrittweise Ausführung, Verzweigungen und Schleifen.
- Handlungsabläufe in ein formales System übertragen.
- Handlungsabläufe in einem formalen System verstehen und sie mit Hilfe von Alltagssprache erklären.
- Fehler in Handlungsabläufen erkennen und beheben.
- die Komplexität eines Handlungsablaufs bewerten.

4.1.4 Informationsrepräsentation in digitalen Systemen

Die Schülerinnen und Schüler können...

- grundlegende Repräsentationsformen von Zahlen in digitalen Systemen verwenden, insbesondere Darstellungsformen für ganze Zahlen und Dezimalzahlen.
- die Grenzen der Repräsentation von Zahlen in digitalen System benennen und daraus resultierende Probleme exemplarisch benennen.
- zusammengesetzte Repräsentationsformen wie Listen und Matrizen verwenden.

4.2 Praktische Umsetzung

4.2.1 Medium

Um die oben genannten Kompetenzen zu erwerben braucht es ein beliebiges formales System, das von einer Maschine interpretiert werden kann. Dieses Medium kann unter anderem eines der folgenden sein:

- ein programmierbarer Taschenrechner
- eine klassische imperative Programmiersprache wie z. B. Pascal oder Logo
- eine moderne Programmiersprache wie Java, C#, Python, Ruby oder Scala.

4.2.2 Unterricht

Vor dem Hintergrund einer offensichtlichen thematischen Überlappung zwischen Mathematik und Informatik im Bereich der Algorithmik wäre es wünschenswert, dass der Teil Algorithmik von einer Mathematiklehrperson unterrichtet wird, die genügend Hintergrundwissen in Informatik hat oder gar Informatiklehrperson ist. Aus diesem Grund hat die Fachschaft Informatik der Kantonsschule Zug (2010) diese Stunde Algorithmik auch vom „normalen" Mathematikunterricht organisatorisch getrennt. Damit wäre auch eine Begrenzung auf eine Jahreswochenstunde sicher gestellt. Zudem ist die Vorgabe, dass der Unterricht in Algorithmik im Abteilungsunterricht stattfinden soll (Arbeitsgruppe Porträt/Stundentafel/Lehrplanarbeit des Stufenprojekts Gymnasien, 2012, S. 3) so einfach umzusetzen.

5 Fazit

Die Behandlung von Algorithmen im Mathematikunterricht ist aus mathematikdidaktischer Sicht notwendig und über den Inhalt selbst hinaus nützlich. Sie kann bestimmten Schülerinnen und Schülern den allgemeinen Zugang zur Mathematik in einer Form erleichtern, die sonst in der Mathematik schwer zu finden ist, und darüber hinaus neue Chancen für den Mathematikunterricht bieten.

Die Mathematik selber ist reich an Beispielen, in denen nicht-triviale Algorithmen explizit oder implizit verwendet werden. Diese kommen zudem in allen Teilgebieten der Mathematik vor.

Ausgehend vom Lehrplan des entsprechenden kantonalen Zusatzfaches im Kanton Zug werden mit „Problemanalyse“, „Verfahren zum strukturierten Lösen von Problemen“, „Handlungsabläufen“ und „Informationsrepräsentation in digitalen Systemen“ vier Inhaltsbereiche präsentiert und in Kompetenzen formuliert, die für den kantonalen Lehrplan in Basel-Stadt eine geeignete Umsetzung dieser Erkenntnisse bieten. Einige Vorschläge zur praktischen Umsetzung sollen helfen, dass die Umsetzung auch im Hinblick auf die organisatorischen Gegebenheiten stattfinden kann.

6 Literaturverzeichnis

Arbeitsgruppe Porträt/Stundentafel/Lehrplanarbeit des Stufenprojekts Gymnasien (2012): *Erläuterungen zur Stundentafel Gymnasien*. URL: http://www.schulharmonisierung-bs.ch/paedagogik/lehrplaene-und-stundentafeln/Erlaeuterungen%20zur%20Stundentafel%20Gymnasien.pdf/at_download/file.

Arbeitskreis Bildungsstandards der Gesellschaft für Informatik e.V. (2008): *Grundsätze und Standards für die Informatik in der Schule – Bildungsstandards Informatik für die Sekundarstufe I. LOG IN*, 150/151. Beilage.

Baumann, Rüdeger (2011): *Auf dem Weg zu Bildungsstandards Informatik für die Sekundarstufe II – Probleme und Lösungsvorschläge. LOG IN*, 169/170, S. 60–71.

Bohnacker, H., Gross, B., Laub, J. und Lazzeroni, C. (2012): *Generative Design: Visualize, Program, and Create with Processing*. Princeton Architectural Press.

Borwein, Jonathan, Devlin, Keith und Girgensohn, Roland (2010): *Experimentelle Mathematik: Eine beispielorientierte Einführung*. Spektrum Akademischer Verlag, Heidelberg, Berlin.

Cohors-Fresenborg, Elmar und Kaune, Christa (1993): *Zur Konzeption eines gymnasialen mathematischen Anfangsunterrichts unter kognitionstheoretischem Aspekt. Der Mathematikunterricht*, 39(3), S. 4–11.

Fachschaft Informatik der Kantonsschule Zug (2010): *Informatik – Lehrplan für das kantonale Zusatzfach*. URL: http://www.zug.ch/behoerden/direktion-fur-bildung-und-kultur/ksz/dokumente/lehrplaene/download-kantonale-zusatzfaecher/informatik-kantonales-zusatzfach/at_download/file_pdf.

Gressley Freimann, Philipp und Guggisberg, Martin (2011): *Programmieren lernen – Aufgaben für den Informatikunterricht – Sekundarstufe II*. Orell Füssli, Zürich.

Klafki, Wolfgang (1994): *Neue Studien zur Bildungstheorie und Didaktik: zeitgemässe Allgemeinbildung und kritisch-konstruktive Didaktik*. Beltz, Weinheim, Basel, 4. Auflage.

Leuders, Timo, Hg. (2003): *Mathematik-Didaktik – Praxishandbuch für die Sekundarstufe I und II*. Cornelsen Scriptor, Berlin.

Oldenburg, Reinhard (2011): *Mathematische Algorithmen im Unterricht: Mathematik aktiv erleben durch Programmieren*. Vieweg und Teubner, Wiesbaden.

Prusinkiewicz, Przemyslaw und Lindenmayer, Aristid (1990): *The Algorithmic Beauty of Plants*. Springer Verlag, New York NY.

Rozenberg, Grzegorz und Salomaa, Arto (1980): *The Mathematical Theory of L-Systems*. Academic Press, New York NY.

Schubert, Sigrid und Schwill, Andreas (2004): *Didaktik der Informatik*. Spektrum Akademischer Verlag, Heidelberg, Berlin.

Schwank, Inge (1993): *Verschiedene Repräsenationen algorithmischer Begriffe und der Aufbau mentaler Modelle*. *Der Mathematikunterricht*, 39(3), S. 12–26.

Sjuts, Johann (1999): *Mathematik als Werkzeug zur Wissensrepräsentation – Theoretische Einordnung, konzeptionelle Abgrenzung und interpretative Auswertung eines kognitions- und konstruktivismustheoriegeleiteten Mathematikunterrichts*, Band 35 von *Schriftenreihe des Forschungsinstituts für Mathematikdidaktik*. Forschungsinstitut für Mathematikdidaktik, Osnabrück.

Strauch, Dietmar und Rehm, Margarete (2007): *Lexikon Buch, Bibliothek, neue Medien*. Saur, München, 2. Auflage.

Ständige Konferenz der Kultusminister der Länder in der Bundesrepublik Deutschland (2012): *Bildungsstandards im Fach Mathematik für die Allgemeine Hochschulreife*.
URL: http://www.kmk.org/fileadmin/veroeffentlichungen_beschluesse/2012/2012_10_18-Bildungsstandards-Mathe-Abi.pdf.

Terzidis, Kostas (2009): *Algorithms for Visual Design Using the Processing Language*. Wiley Publishing.

Tietze, Uwe-Peter, Klika, Manfred und Wolpers, Hans, Hg. (2000a): *Mathematik in der Sekundarstufe II – Band 1: Fachdidaktische Grundfragen – Didaktik der Analysis*. Vieweg, Braunschweig, Wiesbaden, 2. Auflage.

Tietze, Uwe-Peter, Klika, Manfred und Wolpers, Hans, Hg. (2000b): *Mathematik in der Sekundarstufe II – Band 2: Didaktik der Analytischen Geometrie und Linearen Algebra*. Vieweg, Braunschweig, Wiesbaden.