Informatik-Schnuppereinheiten zur Positionierung des Unterrichtsfachs

Ernestine Bischof, Roland T. Mittermeir

Institut für Informatik-Systeme
Universität Klagenfurt
Österreich
{ernestine, roland}@isys.uni-klu.ac.at

Abstract. Informatikunterricht im allgemeinbildenden Unterricht verfehlt heute sehr oft das Ziel, Begeisterung für dieses Fach zu wecken. Im Gegenteil, zu starke Fokussierung auf Anwendungsschulung vermittelt ein falsches Bild von Informatik und führt dazu, dass viele an sich Qualifizierte, vor allem qualifizierte Mädchen, als künftige Informatikstudenten verloren gehen.
Das hier vorgestellte Konzept aufbauender Schnuppereinheiten versucht, diesem Trend entgegenzuwirken. Ziel ist, Kinder und Jugendliche in altersgerechter Form an Thematiken des technischen Fachs *Informatik* heranzuführen und ihnen so zu zeigen, dass sich weitere Beschäftigung mit Fragestellungen der Informatik intellektuell lohnt.

1 Motivation

Als Informatik als Schulunterrichtsfach eingeführt wurde, war es für viele Jugendliche ein faszinierendes Fach. Es galt als modern und zukunftsgerichtet, wenngleich nicht immer leicht. Dieser Befund gilt, wie in persönlichen Gesprächen sowie in gezielten Interviews [AKLU07] und umfangreichen Befragungen [Mich08] feststellbar, heute nicht mehr. Der Grund dafür mag zu einem guten Teil darin liegen, dass Computer die Aura des Besonderen verloren haben und in Form des Personal Computers zu Gegenständen unseres täglichen Lebens geworden sind. Zu einem weit stärkeren Teil mag aber Form und Inhalt des Unterrichts, den Kinder und Jugendliche unter der Überschrift Informatik vermittelt bekommen, für diese Umkehr der Wertigkeit verantwortlich sein.

War Informatikunterricht in seinen Anfängen weitestgehend durch das Lösen von Programmieraufgaben geprägt [Reit05], so dominiert heute wenigstens an vielen österreichischen Schulen ein an den Ausbildungszielen der European Computer Driving Licence (ECDL) orientiertes Curriculum. Dieser Wechsel kann begründet werden; etwa dadurch, dass nicht eine ganze Population Jugendlicher zu Software-Entwickler/innen ausgebildet werden soll. Auch gibt die Orientierung am ECDL Lehrkräften, die sich mit einem sehr offenen Lehrplan konfrontiert sehen, eine Handreichung bei der Stoffauswahl. Darüber hinaus erfolgt durch die schulexterne ECDL-Prüfung sowohl eine Zertifizierung des Unterrichts als auch der Unterrichteten.

Allerdings ging dadurch die intellektuelle Faszination am Fach, wie sie etwa von Dagys, Dagiene und Grigas in der Beschreibung der Anfangsphasen des Informatikunterrichts, in der Programmieraufgaben in Zeitungen publiziert wurden und jene, die sie lösten, ihre Programme zur Überprüfung an ein Universitätsinstitut senden konnten [DDG06], verloren. In einer von Micheuz [Mich09] an österreichischen Schulen durchgeführten Befragung umschrieben Jugendliche der 9. Schulstufe die Inhalte des Informatikunterrichts dominant mit *Word*, *Textverarbeitung*, *Excel*, *langweilig*, *Präsentation*. Begriffe wie *spannend* oder *interessant* folgten erst mit deutlichem Abstand. Dabei fällt nicht nur auf, dass Produktnamen vor dem generischen Begriff kommen, sondern vor allem, dass der Unterricht des Texteschreibens offenbar so dominant ist, dass er sogar beide Spitzenpositionen einnimmt. In diesem Kontext sollte jedoch der hohe Rang von „langweilig" oder „uninteressant" nicht sehr wundern. Mögen sich doch all jene, die Maschinschreibunterricht besuchten, an diesen zurückerinnern. War dieser intellektuell anspruchsvoll? Hätte er getaugt, für ein anschließendes Maschinschreibstudium zu motivieren?

Wir haben mithin erkannt, dass der an vielen österreichischen Schulen mit bester Absicht praktizierte Informatikunterricht kaum geeignet ist, die kreativsten Köpfe für ein nachfolgendes Informatikstudium zu motivieren. Insbesondere aus einer an zwei Schulen durchgeführten Studie [AKLU07] ging hervor, dass speziell bei Mädchen negative Vorurteile gegenüber Informatik und Informatikfreaks bestehen, die zwar einem Zerrbild entsprechen, jedoch durchaus ernst zu nehmen sind.

Ausgehend von diesen Überlegungen starteten wir ein Projekt, das Fach Informatik im Bewusstsein Jugendlicher dort zu positionieren, wo es sich aus der Eigensicht tatsächlich befindet. Wir wurden dabei durch einen Beirat von Schulpädagog/innen, die in unterschiedlichen Schulstufen tätig sind, begleitet. Insbesondere sollte klar gestellt werden, dass Informatik ein konstruktives technisches Fach ist. Damit sollte die Verwechslung zwischen Fahrschule und Konstruktionsbüro aus der Welt geschaffen werden. Allerdings ist dabei zu berücksichtigen, dass der Unterschied zwischen Fahrschule und Konstruktionsbüro leichter nachvollziehbar ist, als in Informatik. Die unterschiedlichen Fahrzeugtypen und Modelle sieht man ja, man kann sie auch ausprobieren und erlebt so wenigstens die Effekte der Tätigkeiten der Konstrukteur/innen. Um Informatik erlebbar zu machen, bedarf es einiger Umwege, die im folgenden Kapitel dargestellt werden.

2 Informatik erLeben – Konzept einer Zurechtrückung

Ausgehend von obigem Befund versuchten wir, ein didaktisches Konzept zu entwickeln, das in Analogie zu Fahrschule – Konstruktionsbüro – (Fertigungsstraße) – Autofahren für den Informatikbereich die Unterschiede zwischen ECDL – Software-Konzeption – (Software-Fabrik) – Software-Anwendung verdeutlicht. Weiters sollte in Burschen wie vor allem auch in Mädchen Interesse am späteren Studium eines technischen Fachs wie Informatik geweckt werden.

Allerdings sind zum Unterschied von Maschinenbau oder Bauingenieurwesen/Architektur die Produkte von Informatikern nicht (Software) oder nicht in ihrem eigentlichen Wesen (Hardware) sichtbar und daher in ihrer Funktionsweise nicht erlebbar. Hilfestellung bei der Überwindung dieser Hürde bot Antonitschs Idee, Kara-Programme [RNH04] auf den Fliesen des Schulgangs durch Schüler/innen nachspielen zu lassen [ALS07]. Diese Animation birgt eine Fülle von Implikationen:

- Das Konzept des Automaten als beschränktes, strikt Anweisungen befolgendes Gerät wird durch die Kara simulierende Person deutlich. Es wird insbesondere dann deutlich, wenn diese Person versucht, intelligenter zu sein, als die eben erhaltene Anweisung und die Beobachter der Szene auf das Missverhältnis zwischen Programm und Handlung aufmerksam werden.
- Es wird klar, dass *Programmieren* keine magische Tätigkeit ist, sondern einfach darin besteht, exakt zu formulieren und künftige Situationen in den niedergeschriebenen Verfahrensanweisungen zu antizipieren.
- Ebenso wird klar, dass es, um die Prinzipien des *Programmierens* zu verstehen und zu erlernen, keiner besonderen, irgendwie kryptisch erscheinenden Programmiersprache bedarf. Wir müssen uns nur einer Sprache bedienen, die der Adressat unserer Anweisungen, sei es Mensch oder Maschine, versteht.
- Das Verfahren wird von der „programmgesteuerten Person“ so langsam ausgeführt, dass die gesamte Klasse den Vorgang im wahrsten Sinn des Wortes Schritt für Schritt nachvollziehen kann.
- Letztlich konnte festgestellt werden, dass diese Einheiten den Jugendlichen Spaß bereitet haben und das ist bei gutem Unterricht ja nicht verboten.

Aus obigen Punkten wird klar, dass derartige Interventionen in den Informatikunterricht zweifellos allgemeinbildenden Charakter haben. Schärfung des Ausdrucks, Berücksichtigung der sprachlichen (und gegebenenfalls auch intellektuellen) Fähigkeiten des „Kommunikationspartners“, Antizipation von Situationen, etc., sind Fähigkeiten, die in der Informatik wie im Leben jedes Menschen wichtig sind. Sich darin zu üben hat also wenig mit künftigen Informatikberufen zu tun. Es ist aber durchaus anspruchsvoll und ist auf einer anderen als auf der eher sportlichen Skala zu bewerten, die misst, wie viel Nettoanschläge pro Minute man in Maschinschreiben schafft oder wie viel Word- und/oder Excel-Befehle man auswendig kann. Simulation und Spiel stellten mithin eine Leitlinie für unser Projekt *Informatik erLeben* dar.

Eine weitere Rahmenbedingung wurde uns von den Schulpraktikern auferlegt. Sie meinten, man müsse mit einem derartigen Projekt auf Volksschulebene beginnen, da im Alter von acht bis zehn Jahren noch ausreichend Begeisterungsfähigkeit vorläge und somit wesentliche Weichen für den späteren schulischen wie außerschulischen Bildungsverlauf gelegt würden. Dies schraubte die Hürde für unser Vorhaben noch etwas weiter in die Höhe, da dadurch der noch beschränkte Vorrat an mathematischem Vorwissen, und die noch beschränkte Abstraktionsfähigkeit der Zielgruppe berücksichtigt werden mussten.

Schlussendlich waren wir der Meinung, dass eine Intervention auf Volksschulebene wenig bringt, wenn nicht eine weitere, wenn auch nur punktuelle, Betreuung über die restliche Schulzeit aufrecht erhalten werden könnte.

Auf dieser Grundlage erarbeiteten wir einige Themenstränge, in denen Informatikinhalte (oder auch allgemeinere Technikinhalte) so dargestellt werden konnten, dass sie in altersgerechten Schritten Technik- und Informatikverständnis vermitteln. Der Beirat wählte dabei insbesondere jene Themen aus, die für die oberen Grundschulklassen (in Österreich 3. und 4. Klasse Volksschule) geeignet erschienen. Da sich einige dieser Themen nicht für eine durchgängige Fortsetzung bis zu Abiturklassen anboten (z.B. Bilder und Farben) oder am Weg dorthin Lücken aufwiesen (z.B. Verschlüsselung), ergänzten wir die Themen noch um einige Gebiete, die uns fachlich besonders am Herzen lagen. Aus diesen Überlegungen resultierten die in Kapitel 4 beschriebenen neun Themenstränge.

Die für die Intervention in Schulen vorgesehenen Einheiten umfassen jeweils zwei Schulstunden (oder 100 Minuten). Die meisten Einheiten sind weiter gegliedert in Module. Dadurch können auch themenstrangübergreifende Topoi besprochen werden. Einige didaktische Grundüberlegungen sollten dabei beachtet werden:

- Insbesondere für die noch sehr junge Zielgruppe sollte wenigstens einer der Module stark spielerischen Charakter haben. Aber auch ältere Jugendliche sollten etwas mitnehmen können, mit dem sie in einer Peer-Gruppe beeindrucken können (Etwa: „Wie weit kannst Du mit den Fingern beider Hände zählen?“ – „Ich schaffe es bis über 1000!“)
- Wo immer möglich sollten Verbindungen zur Erfahrungswelt der Kinder und Jugendlichen aufgebaut werden.
- Animationsanteile der Module sollten zur Beobachtung anregen. Schulung des Beobachtungsvermögens sollte Beobachtungswillen und Neugierde stimulieren.
- Die Einheiten enthalten viele Fragen an die Klasse. Weitere Fragen sollten angeregt werden.
- Dass Informatik (und Technik allgemein) auf naturwissenschaftlichen Grundlagen wie Mathematik und Physik beruht, wird nicht verschwiegen. Im Gegenteil! Aber diese Grundlagen müssen so einfach gebracht werden, dass sie für die Kinder bzw. Jugendlichen der jeweiligen Altersgruppe nachvollziehbar sind.
- Dabei, und natürlich auch bei den Informatikinhalten im engeren Sinn, darf allerdings nie auf Korrektheit verzichtet werden. Diese schwierige Randbedingung muss durch Wahl des richtigen Abstraktionsniveaus eingehalten werden. Analogien wirken unterstützend, dürfen aber nie überzogen werden. Das Feld, aus dem die Analogien stammen, ist weit. Es reicht von Kunst bis zur Natur.
- Unter den Fragen sollten auch solche sein, die schon einiges an Mit- und Nachdenken verlangen. Aber wir wollen sicherlich keinen Prüfungsstoff vermitteln.
- Wenn dies möglich ist, werden kontrastierende Verfahren vorgestellt, um zu zeigen, dass sich Nachdenken lohnt und dass Informatik kein naives Bastel- und Probierfach ist, auch wenn das Experiment einen wichtigen Stellenwert hat.
- Auch sollte klar gestellt werden, dass Informatikerinnen und Informatiker nicht die Maschine zum Ziel ihrer Tätigkeit haben, sondern dass die Maschine vielmehr ein Instrument ist, das im Rahmen eines Gesamtkonzepts in einer für andere Menschen sinnvollen und effizienten Weise eingesetzt wird.

Aus diesen Randbedingungen folgt, dass hier kein abgeschlossenes, umfassendes didaktisches Konzept realisierbar ist, sondern nur lose aufeinander aufbauende, aber in sich abgeschlossene Interventionseinheiten angeboten werden.

4 Verfügbare Einheiten

Die Umsetzung dieser Ziele erfolgt in neun Themensträngen, die in einzelne Einheiten bzw. Module gegliedert sind. An einigen Stellen bestehen explizite Querbeziehungen, sodass auch Einheiten aus verwandten Modulen unterschiedlicher Stränge (etwa Binärsystem und Addierwerk) zusammengestellt werden können.

4.1 Bilder, Grafik und Zeichnen

Dies mag nicht als zentrales Thema der Informatik erscheinen. Da die ins Auge gefassten Zielgruppen aber schon im Grundschulbereich beginnen, bietet es einen Anknüpfungspunkt zum Vorwissen der Kinder und kann gut spielerisch aufgearbeitet werden. Im Modul B1 werden physikalische Grundlagen zur Farbwahrnehmung behandelt. Darauf aufbauend können die Kinder im Modul B2 zur Farbsynthese selbst die Druckfarben Gelb, Magenta und Cyan mischen. Sie lernen so die Unterschiede zwischen additiver und subtraktiver Farbmischung kennen. Ebenfalls bereits ab der Primarstufe empfohlen ist Modul B3 Grafikformate, der erklärt wie Bilder im Computer gespeichert werden können, obwohl dieser nur auf der Basis von 0 und 1 arbeiten kann. Ein weiteres Grundschulmodul ist die Druckausgabe.

Abbildung 1: SchülerInnen der Primarstufe bei einer Informatik-erLeben Piloteinheit

Das Thema Farbtiefen ist für Klassen ab der Sekundarstufe I vorgesehen. Abb. 1 zeigt Schüler/innen einer 3. Volksschulklasse beim Ausmalen von Farbkreisen mit den Druckerfarben. Für uns war diese Einheit wichtig, da sich zeigte, dass aufgrund der noch eingeschränkten Motorik die Vorlagen größere Malflächen erfordern.

4.2 Codierung

Mit der Codierung nähern wir uns bereits den Kernthemen der Informatik. In den Einheiten zur Codierung werden wichtige Verständnisgrundlagen für die Informatik geschaffen. Im Grundschulbereich beginnen wir wieder spielerisch, etwa mit einem „Morsespiel“ in Modul C1. Anhand des Morsecodes wird das Grundprinzip der Zuordnung von Zeichen eines Alphabets zu Zeichen eines anderen Alphabets veran-

Abbildung 2: Fehlerkorrektur mit Prüfbits in der Hauptschule Kühnsdorf (Sekundarstufe I)

schaulicht. Im darauf aufbauenden Modul können die Kinder selbst einen Code mit verschiedenen Farben entwickeln. Anschließend können sie dann ihre Namen im Morsecode und in einem selbst entwickelten Code codieren. In weiteren Modulen werden Codebäume eingeführt, in der Primarstufe noch sehr spielerisch, ab der Sekundarstufe sollen dann auch bereits eigene kleine Codierungen mit Codebäumen entstehen. Codeoptimierung und Huffmann Codierung werden ebenso thematisiert. Das Modul C4 bietet ab der Sekundarstufe I eine Einführung in das Binärsystem basierend auf Ideen in [BWF06].
Fehlererkennung und Fehlerkorrektur werden, wie obiges Foto zeigt, im Modul C6 für die Sekundarstufe spielerisch thematisiert ([BWF06]).

4.3 Verschlüsselung

Mit einfachen Verschlüsselungsverfahren, wie der Cäsar-Verschlüsselung beginnen wir bereits in Grundschulklassen spielerisch einfache Sätze zu ver- und entschlüsseln. Ab der Sekundarstufe II kann man mit den Schüler/innen bereits verschiedene symmetrische Verschlüsselungsverfahren, wieder ausgehend von der Cäsar-Verschlüsselung, vergleichen und ihre Sicherheit abwägen. Da die Verschlüsselung bei der virtuellen Kommunikation immer wichtiger wird, kann hier den Kindern und Jugendlichen auch starker Praxisbezug vermittelt werden. In der Sekundarstufe II werden auch komplexere Verfahren, wie öffentlicher Schlüsseltausch und Public-Key-Verschlüsselungsverfahren (RSA) thematisiert.

4.4 Hardware

Der Computer wird in der Schule und von unterschiedlichen Berufsgruppen im Alltag verwendet. Was nun aber wirklich in einem Computer drinnen steckt und wie er rechnet, wissen nur wenige. In den Einheiten H1 bis H3 sollen die Schüler/innen altersgerecht die grundlegende Rechenweise des Computers kennen lernen. In Animationen übernehmen sie die Funktion der Computerbauteile. Im Grundschulbereich berechnen sie einfache Additionen oder Sub-

Abbildung 3: Volksschülerin mit Motherboard

traktionen, später komplexerer Rechenabfolgen in einer vereinfachten Maschinensprache. Um zusätzlich zum Verständnis auch noch ein Bild von der Hardware zu vermitteln, werden im Anschluss nicht mehr gebrauchte Rechner geöffnet und die einzelnen Bauteile benannt. Im Modul H4 stellen die Schüler/innen die Funktionsweise eines Halb- und Volladdierers dar.

Abb. 3 zeigt eine Schülerin einer 4. Volksschulklasse mit einem Motherboard. Vorher spielten die Kinder mit einfachen Rechenaufgaben die Funktionsweise einer CPU. Besonderes Interesse erweckte auch eine zerlegte Hard-Disk.

4.5 Netze

Der vielfältige Bereich der Computernetze wird für Grundschulkinder spielerisch, etwa mit einem „stille Post"-Spiel aufgearbeitet. Wie es bei der Kommunikation zwischen Menschen Höflichkeitsformen und Kommunikationsregeln gibt, so gibt es beim Nachrichtenaustausch zwischen Computern auch Protokolle, die gewisse Regeln festlegen. Von diesen elementaren Grundlagen kommen wir dann in der Sekundarstufe zu den Netzwerkprotokollen und Netzwerktypen. Was passiert aber, wenn wir im Auto mit dem Mobiltelefon telefonieren oder im Internet surfen? Wie bleibt die Verbindung aufrecht? Diese Themenbereiche werden mit Schüler/innen der Sekundarstufe II besprochen.

4.6 Betriebssysteme

Vielen Anwendern ist der Unterschied zwischen Betriebssystem und Anwenderprogrammen nicht wirklich bewusst. Einige wissen, dass beide zur Software zählen, andere wissen nicht einmal, dass man zwischen den beiden unterscheidet. Die zentrale Rolle des Betriebssystems wird für Kinder der Primarstufe mit einer Apothekerin (Betriebssystem) verglichen, die Kunden (Prozesse, Anwenderprogramme) bedient. Sie verteilt die Ressourcen, im Computer sind dies etwa die CPU, Speicher oder der Drucker, an die einzelnen Anwendungsprogramme. Ab der Sekundarstufe gehen wir dann auch bereits auf Scheduling-Strategien bei der Vergabe dieser Ressourcen ein, sowie auch auf mögliche Probleme wie Deadlocks oder Lifelocks.

4.7 Programmieren

Programmieren im Grundschulalter, und das auch noch ohne Computer? Wir beginnen hier im Modul P1 mit einfachen Wegbeschreibungen, etwa innerhalb des Schulgebäudes, welche die Kinder in freiem Text verfassen. Um auf die Bedeutung von Präzision in solchen Beschreibungen/Algorithmen aufmerksam zu machen, probieren die Kinder ihre Beschreibungen gegenseitig aus. So können Fehler oder Ungenauigkeiten im Algorithmus entdeckt werden. Soll ein Algorithmus für einen Computer ausführbar sein, muss die Beschreibung exakt und eindeutig sein. So sollen auch bereits kleinere Kinder zum Algorithmus-Begriff hingeführt werden, allerdings noch ohne zu Programmieren. Im Modul P2 sollen die Wegbeschreibungen bereits mithilfe

einfacher Kontrollstrukturen wie Folgen, Alternativen (Wenn – Dann) oder Schleifen (Wiederholung) strukturiert werden. Die Module P3 und P4 behandeln unterschiedliche Sprach- und Kommunikationsstile. Diese Module können in höheren Klassen als Einstieg in die Programmierung bzw. in Objektorientierung verwendet werden. In Modul P5 wird das grundlegende Denkmuster der Rekursivität anhand einfacher Beispiele aus Kunst, Natur, Sprache und Mathematik vermittelt.

4.8 Sortieren

In den Einheiten und Modulen zu Sortieralgorithmen sollen die Jugendlichen Einblick bekommen, wie nützlich das Sortieren von Daten ist. Um aber überhaupt einmal einige Sortierverfahren kennen zu lernen, spielen die Schüler/innen das Verfahren in einer Animation durch. Sie stellen sich in einer ungeordneten Reihe auf und werden von einer weiteren Person dann etwa nach dem Geburtsdatum sortiert. Sie sollen dazu hingeführt werden, verschiedene Verfahren nach ihrer Geschwindigkeit bzw. Effizienz zu vergleichen. Neben Bubblesort (So1) und Selectionsort (So2), die auch bereits in der Grundschule ansatzweise vorgestellt werden können, werden auch komplexere Verfahren wie Mergesort (So3) und Quicksort (So4) vorgestellt und bewertet.

In einer Übungseinheit entwickelten die Teilnehmer selbständig ein an Radixsort angelehntes Verfahren. Dies zeigt, dass es sich bei all diesen Interventionen lohnt, mit Fragen zu beginnen und mit dem vorbereiteten Konzept recht offen umzugehen. Manchmal ergeben sich spontan Situationen, die exzellent in das didaktische Konzept passen und so die Motivation erhöhen. Freilich zeigt eben dieses Beispiel, dass die Systematisierung des Vorschlags nach Kategorien innerhalb von Rechnungsnummern zu sortieren für einige Teilnehmer/innen nicht mehr ganz nachvollziehbar war.

4.9 Suchen

Suchalgorithmen gehören, wie Sortieren, zu den Grundlagen der Informatik und stellen ein zentrales Thema dar. Suchstrategien hat bereits jeder von uns unabhängig von informatischen Strukturen entwickelt, sei es ob man in einem Telefonbuch sucht oder in ungeordneten Strukturen. Daher wird mit elementaren Suchstrategien bereits in der Grundschule begonnen, etwa im Modul Su1 mit der blinden Suche in einem verschlossenen Sack. Die Kinder sollen intuitiv eine Suchstrategie anwenden und diese dann für die anderen Kinder beschreiben. Auf den ansteigenden Schwierigkeitsgrad diese Aufgabe auszuführen und dann ihre Lösung anschließend zu verbalisieren wird in [Kol08] hingewiesen. Diese Übung eignet in Kombination mit den Moduln P1 oder P2 auch als Vorbereitung auf Programmierung.

Darauf aufbauend soll in den folgenden Modulen erkannt werden, wie der Suchaufwand von der Struktur der Daten abhängt, etwa ob wir eine unsortierte oder eine sortierte Struktur vor uns haben. Um Praxisbezug herzustellen, kann auf Suche nach kritischen Messdaten hingewiesen werden. Hier kann ein schneller Treffer lebensrettend sein.

5 Wege zum Ziel

Wir sehen zwei Wege vor, die eben beschriebenen Interventionseinheiten an Schülerinnen und Schüler heranzutragen.

Der erste Weg ist, dass wir anbieten, selbst oder durch Kolleginnen, die die Entwicklung der Einheiten verfolgten und wohlmeinend kritisierten, Schulbesuche abzustatten, in denen wir mit der Klasse eine zweistündige Unterrichtssequenz gestalten. Dies erfolgt nach vorhergehender Rücksprache mit der Informatiklehrkraft oder der Klassenlehrkraft (in Volksschulen gibt es in Österreich keinen gesondert ausgewiesenen Informatikunterricht, aber im Sachunterricht kann auf Informatik Bezug genommen werden). Diese Lehrkraft kennt die Klasse und bestimmt daher das Thema der konkreten Intervention.

Um Lehrkräfte auf diese Möglichkeit aufmerksam zu machen stellten wir diese Option in dem Heftchen *Informatik erLeben* [BiMi08] vor. Es konnte Dank einer Förderung des Kärntner Wirtschaftsförderungsfonds und des IT-Campus allen einschlägigen Lehrerinnen und Lehrern Kärntens zur Verfügung gestellt werden. Dieses Heftchen sollte jedoch aufgrund einer dringenden Empfehlung des Beirats kurz gehalten sein. Es enthält daher neben einer Vorstellung des Projekts lediglich 5 Module (zwei zu Codierung, einer zu Verschlüsselung und zwei kontrastierende Sortierverfahren). Die übrigen Einheiten werden schrittweise auf der Web-Site http://informatik-erleben.uni-klu.ac.at publiziert.

Der zweite Weg führt teils über den ersten, teils wird er direkt angestrebt. Eine Grundidee des eben beschriebenen Wegs ist, dass Lehrkräfte unser Wirken in der Klasse beobachten und künftig die Einheiten, die wir mit den Kindern bzw. Jugendlichen übten, selbst durchführen. Dieser Weg kann natürlich auch direkt beschritten werden. Die von uns verwendeten Materialien sind in der Regel billig, teils sind sie über die oben angegebene Web-Site beziehbar, teils werden sie, wie etwa ein veralteter Computer, den man zerlegen und wieder zusammenbauen kann, an der Schule ohnehin vorhanden sein.

Dieser zweite Weg bereitete uns allerdings etwas mehr Arbeit als ursprünglich geplant. Die Ausarbeitungen der Module und Einheiten waren ursprünglich so, dass wir und wohl auch jede/r andere Absolvent/in eines Lehramtsstudiums Informatik anhand dieser Skizzen den vorgezeichneten Unterricht abhalten könnte. Hier sprang der Beirat allerdings wieder warnend ein. Nur wenig Informatik Unterrichtende sind Absolventen einer einschlägigen Studienrichtung. Daher sind die Materialien zwar als Lehrbehelfe zu gestalten, doch dies hat so zu erfolgen, dass sie gleichzeitig auch selbst die Rolle von Lehrmaterialien für Lehrende übernehmen können.

Zum Glück hilft uns dabei die Technologie, die es zu vermitteln gilt. Da wir viele der Einheiten schon mit Kindern erprobt haben, konnten wir uns über deren Angemessenheit Klarheit verschaffen. Für die bereitgestellten Materialien können wir dies noch nicht behaupten. Da sie jedoch via Web-Site publiziert sind, muss man bei

allfälligem Ergänzungsbedarf nicht auf eine Neuauflage der Sammlung warten. Nötige Aktualisierungen oder Korrekturen können fallbezogen erfolgen.

6 Erfahrungen bei bisherigen Schulbesuchen

6.1 Eindruck aus den Schulen

Viele der vorgestellten Einheiten wurden im Sommersemester 2008 bereits in einer Pilotphase an Schulen durchgeführt. An der Pilotphase waren 3 Grundschulklassen, 3 Klassen aus der Sekundarstufe I, sowie 3 Klassen der Sekundarstufe II beteiligt. Es zeigte sich grundsätzlich positives Feedback und großes Interesse seitens der Schulen. Diese Phase diente allerdings auch zur Verbesserung der Einheiten und zum Einholen konstruktiver Kritik. So zeigte sich etwa, dass bei Einheiten im Sekundarstufenbereich teilweise kleine inhaltliche Vorbereitungen durch den Klassenlehrer nötig sind. Besonders im Grundschulbereich merkten wir auch, dass die Haltung der Klassenlehrerin sowie auch die Motivation der Kinder durch die Klassenlehrerin Einfluss auf das Gelingen oder Misslingen einer Informatik-erLeben-Einheit haben.

Weiters war es sehr wichtig, nicht starr am Konzept der Einheit festzuhalten, sondern bei Bedarf spontan zu reagieren, wenn man merkt, dass die Kinder oder Jugendlichen nicht mehr aufnahmefähig sind. So war es einmal nötig die Einheit abzubrechen und zur Auflockerung ein neues Thema zu beginnen. Grundsätzlich hat sich gezeigt, dass Kinder vor allem bis zur Mitte der Sekundarstufe I sehr begeisterungsfähig sind. Aktive Beteiligung ist allerdings auch in Einheiten für höhere Schulstufen nötig.

Überraschend und erfreulich waren Erfahrungen, die das Interesse der Schüler/innen zeigten. So begann beispielsweise ein Schüler der 5. Schulstufe während der Verschlüsselungseinheit, selbst eine Verschlüsselungsart zu entwerfen, indem er Buchstaben durch Zeichen ersetzte.

5.2 Schüler/innenrückmeldungen

Um auch das Feedback der Schüler/innen einzuholen, wurde in den Grundschulklassen und in einer Klasse der Sekundarstufe I ein Fragebogen ausgegeben. Von den insgesamt 59 Schüler/innen der Grundschulklassen gaben 53 an, dass sie das vorgestellte Thema interessiert hat. 47 gaben an, dass sie sich im Allgemeinen für Technik interessieren. Auf die Frage, ob die *Informatik-erLeben*-Einheit das Interesse an Technik geweckt habe, antworteten 46 Schüler/innen mit „ja“ bzw. „eher ja“. Fairerweise muss man dabei allerdings einen Höflichkeitsbias unterstellen.

Bei den beiden abschließenden Fragen konnten die Schüler/innen in offenen Antworten noch notieren, was sie besonders interessant fanden. Die nachfolgende Tabelle fasst die Ergebnisse aus den drei Grundschulklassen, unterschieden nach dem Thema der Einheit, zusammen:

Bilddarstellung/Grafik	Hardware	Codierung
- Die bunten Farben (6 mal) - Das Malen (4 mal) - Pixelgrafik – Vektorgrafik (4 mal) - Das Mischen der Farben (2 mal) - Das zum Fenster schauen [Anm. Spektroskop] - gar nichts	- Alles war interessant (5 mal) - Das mit dem Computer auseinander nehmen (5 mal) - Für mich war das interessant, wo wir die Kabeln heraus nehmen durften (2 mal) - Die Festplatte (2 mal) - Die Hauptplatine (2 mal) - Der Kühler - Die Eingabe und die Ausgabe - Als ein paar Kinder Arbeitsspeicher gespielt haben - Die Speicherplatte war spannend - Als ich ein Teil vom Computer war - Zum Schluss, als ich das mit den Leitern gefragt habe - Die Striche, die der Professor erklärt hat [Anm. gemeint sind die Leiter am Motherboard] - Nichts	- Das erste Spiel (6 mal) [Morsespiel] - Als wir mit Taschenlampen Signale gegeben haben [ebenfalls Morsespiel] - Unsere eigene Codierung (4 mal) - Der Morse Code (3 mal) - Die Codes und der Rechner des Computers - Alles - Das mit dem Baum [Codebaum] - Der Rechner

Bei Schüler/innen der Sekundarstufe I im Alter zwischen 13 und 15 Jahren ergab sich ein etwas anderes Bild. Sechs der insgesamt 17 Schüler/innen fanden das Thema Codierung interessant, vier wählten „eher ja“ und fünf „eher nein“. Zwei der Schüler/innen fanden das Thema überhaupt nicht interessant.

Das Interesse an Technik war geringer als bei den Grundschülern. Acht der 17 Schüler/innen interessierten sich für Technik, 9 nicht. Auch auf die Frage, ob die Einheit Interesse an der Technik geweckt hat antworteten nur 8 Schüler/innen mit ja. Die Rückmeldungen bei den freien Antworten ergaben:

- Der Morsecode (6 Nennungen)
- Der Codebaum (5 Nennungen)
- Dezimalsystem (2 Nennungen)
- Mir hat alles gefallen
- Alle beide Stunden
- Die Huffman Codierung
- Code erstellen
- Das Spiel

6 Resümee und Ausblick

Die Pilotphase an den Schulen hat gezeigt, dass die vorgestellten Einheiten im Großen und Ganzen für die Zielgruppe angemessen sind und Interesse bei den Schüler/innen wecken können. Die Rückmeldungen der Lehrer/innen sowie der Schüler/innen wurden von uns nebst eigenen Reflexionen in die Einheiten und Module eingearbeitet. Die überarbeiteten Einheiten werden bzw. sind auf der Website des Projektes publiziert, und stehen somit einem breiten Interessentenkreis aller Schulstufen zur Verfügung.

Im Jänner 2009 startete im Rahmen von *Forschung macht Schule* (ein Projekt des Bundesministeriums für Verkehr, Innovation und Technologie (bmvit)) eine dreistufige Kooperation zwischen der Universität Klagenfurt, heimischen Firmen sowie Kärntner Schulen aller Schulstufen. Die Kooperation dauert drei Semester, wobei jede teilnehmende Klasse pro Semester mindestens bei einer *Informatik-erLeben*-Einheit mitmachen wird. Weiters wird jede Klasse eine Firma besuchen, sowie an der Universität einen Blick in die Forschungslandschaft der Informatik wagen.

Literaturverzeichnis

[AKLU07] Antonitsch, P., Krainer, L., Lerchster, R., Ukowitz, M.: Kriterien der Studienwahl von Schülerinnen und Schülern unter spezieller Berücksichtiung von IT-Studiengängen an Fachhochschule und Universität; IFF-Forschungsbericht, Universität Klagenfurt, März 2007.

[ALS07] Antonitsch, P.K, Lassernig, U., Söllei, A.: Lehrarrangements in der Informatiklehrerausbildung. In: Schubert, S. (Hrsg.): Didaktik der Informatik in Theorie und Praxis; Proc. INFOS 2007, LNI 112, Gesellschaft für Informatik, Bonn, 2007, pp. 91 – 99.

[BWF06] Bell, T.; Witten, I. H.; Fellows, M.: Computer Science Unplugged. An enrichment and extension programme for primary-aged children. http://www.google.com/educators/activities/unpluggedTeachersDec2006.pdf (12. 2. 2009).

[DDG07] Dagys, V., Dagiene, V., Grigas, G.: Teaching Algorithms and Programming by Distance: Quarter Century's Activity in Lithuania. In: Dagiene, V., Mittermeir, R. (eds.): Information Technologies at School, Proc. 2nd ISSEP, TEV, Vilnius, 2006, pp. 402 – 412.

[Kol08] Kolczyk, E.: Algorithm – Fundamental Concept in Preparing Informatics Teachers. In: Mittermeir, R.T., Syslo, M.M. (eds.): Informatics Education – Supporting Computational Thinking, Proc. 3rd. ISSEP, LNCS 5090, Springer, 2008, pp. 265 – 271.

[Mich08] Micheuz, P.: Some Findings on Informatics Education in Austrian Academic Secondary Schools. In: Informatics in Education, 2008, Vol. 7(1), pp. 1 – 16; Institute of Mathematics and Informatics, Vilnius.

[Mich09] Micheuz, P.: Theorie, Praxis und Pragmatik informatischer Bildung an der AHS; to appear in Proc. INFOS 2009.

[RNH04] Reichert, R., Nievergelt, J., Hartmann, W.: Programmieren mit Kara: Ein spielerischer Zugang zur Informatik; Springer, 2004.

[Reit05] Reiter, A.: Incorporation of Informatics in Austrian Education: The Project "Computer-Education-Society" in the School Year 1984/85. In: Mittermeir, R.T. (ed.): From Computer Literacy to Informatics Fundamentals, Proc. ISSEP 2005, LNCS 3422, Springer, 2005, pp. 4 – 19.