

Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 1

In der Vorlesung wurde Ihnen exemplarisch anhand einer E-Mail¹ eines Informatiklehrers einer Realschule die Problematik aufgezeigt, im Informatikunterricht in der Sekundarstufe I sowohl „der Motivationslage der Schüler gerecht [zu] werden“ als auch „den grundlegenden Forderungen des Faches [zu] genügen“.

1. Aufgabe: (2 Punkte)

Beschreiben Sie die antizipierte Motivationslage der Schülerinnen eines Wahlpflichtfaches Informatik in der Sekundarstufe I in einer Real- oder Hauptschule.

2. Aufgabe: (2 Punkte)

Zeigen Sie auf, welche „grundsätzlichen Forderungen des Faches“ bestehen.

3. Aufgabe: (2 Punkte)

Welche Erwartungen an den Informatikunterricht bestehen seitens der Eltern, der Schulleitung, ... (bitte weitere Interessensgruppen ergänzen) ?

4. Aufgabe: (2 Punkte)

Wie schätzen Sie den Einsatz von „Standardsoftware“ ein? Geben Sie ggf. an, auf welche Kriterien sie bei ihrer Auswahl besonderen Wert legen oder anderenfalls, warum Sie sich gegen den Einsatz entscheiden.

5. Aufgabe: (2 Punkte)

Beantworten Sie die E-Mail.

Abgabe: 23.10.2003

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ siehe Skript zur Vorlesung „Didaktik der Informatik für die Sekundarstufe I“, Kapitel „1.2 Sekundarstufe I“

Übung "Didaktik der Informatik Sek. I" – Übungsblatt 1 (*)

Aufgabe 1.)

Zur Beschreibung der (vermuteten) Motivationslage von Schülerinnen und Schülern auf einer Real- oder Hauptschule anhand der vorliegenden eMail seien an dieser Stelle zunächst die dort erwähnten Punkte explizit aufgeführt. Herr X Y (der Autor der Mail) nennt in diesem Zusammenhang u.a. folgende Schülerüberungen

"• ich habe zwar einen Computer, kenne aber außer Spielen keine anderen Programme

[...]

• ich brauche meinen Computer später für meine Ausbildung

• ich will bessere Referate und Präsentationen machen

• ich will eine Webseite machen

• ich will lernen, wie man Emails und Briefe schreibt

[...]

• ich will programmieren lernen

• ich will CDs und Filme kopieren können

• ich will mit dem PC Musik machen

• ich will wissen wie man Musik und Filme aus dem Internet holen kann

[...]"

So verschieden (und voneinander unabhängig) diese Aspekte zunächst scheinen, so entspringen sie doch prinzipiell ein und der selben Kategorie: die Verwendung des Computers als Tool, beziehungsweise der Umgang mit dem Computer steht im Vordergrund. Im weitesten Sinne zielen die einzelnen Interessen somit auf die Entwicklung einer "Computer Literacy" ab, wie sie z.B. im Computer Science Unterricht amerikanischer Highschools vermittelt wird.

Eine zweites Feld von Motivationen wird jedoch erkennbar, wenn die in obiger Auflistung ausgelassen Punkte betrachtet werden:

"• ich will alles wissen, was ich brauche, um einen Computer zu kaufen

• ich will meinen Computer selber reparieren können und nicht immer Freunde fragen

• ich will wissen, wie ich mich vor Dialern und Spam schützen kann"

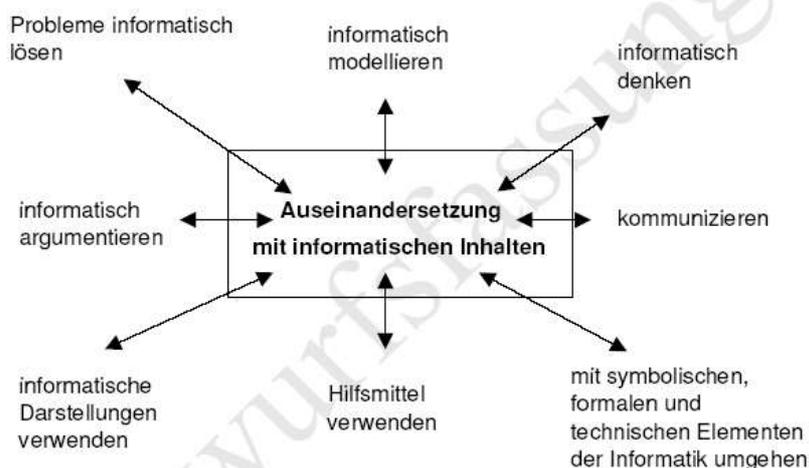
Auch diese Gruppe einzelner Motivation zeigt den Wunsch nach Aufbau einer Computer Literacy, je doch mit anderer Schwerpunktsetzung. Steht bei o.g. Beispielen die Anwendung des Computers (z.B.

mittels Standard-Software) im Vordergrund, so spiegeln die hier erwähnten Punkte den Wunsch nach Wissen ÜBER den Computer wieder – im weitesten Sinne scheint ein technisches Verständnis gewollt. Dies ist sowohl für den Kauf eines Rechners als auch für die Reparatur als auch für die wirkungsvolle Abwehr von Dialern und Spam vonnöten.

An den soeben beschriebenen Schülerwünschen ist jedoch neben der Motivationslage noch ein anderer wichtiger Aspekt erkennbar: große Unterschiede bzgl. des Kenntnisstandes verschiedener Schülerinnen und Schüler. So besteht in einer Äußerung der Wunsch danach, zu "lernen, wie man Emails und Briefe schreibt", in einer anderen Äußerung wird jedoch gleichzeitig um Auskunft gebeten, "wie ich mich vor Dialern und Spam schützen kann.". Ein zusätzliches (und wohlbekanntes) Problem des Informatikunterrichts, das die Notwendigkeit von Differenzierung in den Vordergrund stellt. Dies nur am Rande...

Aufgabe 2.)

Eine konzise Darstellung der "grundsätzlichen Forderungen des Faches" Informatik findet sich auf Seite 18 des Skripts:



Zusammenfassend kann gesagt werden, dass die zentralen Forderungen des Schulfaches Informatik so mit in der Vermittlung dieser Kompetenzen zu sehen sind. Wie in der den Aufgabe zugrundeliegenden eMail ebenfalls erwähnt, sind eben diese Kompetenzen zunächst in der gymnasialen Oberstufe vorhanden und lassen sich Herr Humberts Meinung nach im wesentlichen (d.h. "mit einigen Abstrichen") auf die Sek. I übertragen. Dem möchte ich an dieser Stelle ausdrücklich zustimmen. Es ist enorm wichtig, dass das Fach Informatik nicht zu einem bloßen Software-Anwendungskurs verkommt, sondern dass stets – und eben auch bei der Verwendung von Standard-Software – grundlegende Konzepte begleitend vermittelt werden. – Dazu in späteren Aufgabenteilen mehr.

Aufgabe 3.)

Der besseren Lesbarkeit und Übersicht wegen erfolgt die Beantwortung der Aufgabe als Auflistung:

· **Schulleitung**

Dies hängt nicht zuletzt von der Spezialisierung / dem Schwerpunkt einer Schule ab. So ist es sowohl möglich, dass Informatik als eines von vielen unterrichteten Fächern betrachtet wird als auch, dass informatische Bildung im Mittelpunkt des schulischen Interesses steht.

Wichtigstes Kriterium für die Schulleitung wird sein, ein bestimmtes "Ansehen" der jeweiligen Schule in der Öffentlichkeit herzustellen – somit wird eines der höchsten Anliegen einer Schulleitung sein, dass Inhalte des Informatikunterrichts konform mit den von gesellschaftlichen Institutionen sind. D.h. die Schulleitung hat daran Interesse, die an Sie gestellten Aufgaben zu erfüllen und dadurch das Eingangs erwähnte hohe "Ansehen" zu erreichen. Des weiteren unterliegt eine Schule nicht bloß marginaler Kontrolle durch die zuständige Bezirksregierung, deren Vorgaben ebenfalls erfüllt werden sollen.

· **Arbeitgeber**

Hier besteht vermutlich hauptsächlich der Wunsch nach informatischer Grundbildung in Form von "Beherrschung von Standard-Software". Vorhandenes Wissen auf diesem Gebiet eliminiert die Notwendigkeit, solches Wissen im Rahmen der Ausbildung zu vermitteln und spart somit Geld und Zeit, die sonst für entsprechende Lehrgänge verwendet werden müssten. Selbstverständlich gilt das soeben Gesagte NICHT für informatische Berufsfelder (z.B. Fachinformatiker), da hier sicherlich grundlegendere Kenntnisse informatischer Modelle, Methoden und Prinzipien auf Schülerseite erwartet werden.

· **Gesellschaft**

Dies sei exemplarisch an einem Beispiel erläutert: auf nahezu allen Werbeplakaten, in allen Anzeigen sowie Büchern, usw. finden sich heutzutage URLs der jeweiligen Firmen. Die Fähigkeit zum Aufruf von Websites wird als grundlegende Qualifikation vorausgesetzt, um mehr über Produkte bzw. deren Anbieter zu erfahren. Kenntnisse im Umgang mit Computern sind heutzutage nahezu unabdingbar, um viele Angebote der "Informationsgesellschaft" nutzen zu können. – Mit Leichtigkeit lässt sich dieses Beispiel auf andere gesellschaftliche Bereiche übertragen (z.B. Freunde & Bekannte, die statt Ansichtskarten eMails aus dem Urlaub verschicken)

· **(Lehrer) andere(r) Fächer**

Diese Personengruppe (sofern sie die Nützlichkeit von Computern für ihr jeweiliges Fach erkannt hat) könnte erwarten, dass Schülerinnen und Schüler die im Informatikunterricht erworbenen Kenntnisse in anderen Fächern zum Einsatz bringen kann. Entweder zielt dies ebenfalls auf die Benutzung von Standard-Software ab (z.B. Powerpoint zum Erstellen von Referaten) oder aber auf die Verwendung des Computers als Lernressource (z.B. zur Selbstständigen Arbeit von Schülern unter Zuhilfenahme des Internet als "Nachschlagewerk" und anderweitiger Wissensquelle) – des weiteren ist der Wunsch vorstellbar, dass Schüler Methoden der Informatik auf andere Fächer übertragen können (offensichtlich ist dies z.B. in der Mathematik, da etwa eine FOR-Schleife im weitesten Sinne mit einer Summenformel vergleichbar ist).

Eltern

Die Institution "Eltern" möchte ich hier als Essenz der oben aufgeführten Gruppen darstellen. Sie werden oft in erster Linie ein (begründetes) Interesse daran haben, dass ihr Nachwuchs sich problemlos in einer modernen Gesellschaft zurechtfinden kann. Somit sind sie ebenfalls der Ursprung von Forderungen an die Schulleitung hinsichtlich der Gestaltung eines Faches und beurteilen mittels ihrer Wahl der Schule implizit deren Entscheidungen bzgl. Inhalt der einzelnen Fächer (so auch Informatik). Ebenfalls schließt das Interesse der Eltern eine zufriedenstellende Ausbildung und ein reibungsloser Übergang ihrer Kinder ins Berufsleben mit ein.

Ein weiteres Interesse der Elternseite könnte jedoch sein, selbst etwas über die Benutzung des Computers zu lernen. Somit würden ihre eigene "Unkenntnis" Quelle von Wünschen hinsichtlich informatischer Ausbildung. Teil dieses Interesses ist hier sicherlich wiederum die "Beherrschung" des Computers, d.h. der Umgang mit Standard-Software und das Verständnis von grundlegenden technischen Abläufen (z.B. Notwendig zur Installation neuer Software).

Aufgabe 4.)

Grundsätzlich ist m.E. gegen den Einsatz von Standard-Software im Informatikunterricht nichts einzuwenden – sofern dieser Einsatz dem Verständnis grundlegender Konzepte dient! Es lassen sich somit nämlich "zwei Fliegen mit einer Klappe schlagen": erstens findet eine Vermittlung von Konzepten über die bloße Beherrschung einer einzelnen Anwendung hinaus statt und zweitens erlernen die Schülerinnen und Schüler (entsprechend den o.g. Interessen z.B. eines Arbeitgebers) somit den Umgang mit Software, auf die sie mit großer Wahrscheinlichkeit in kommenden Jahren häufiger treffen werden.

Der erste dieser beiden Punkte kann exemplarisch am Beispiel der Bedienung einer Tabellenkalkulation wie Microsoft Excel erläutert werden. Es ist klar erkennbar, dass bei guter und realitätsnaher (d.h. mit der realen Lebenswelt entnommenen Beispielen) Planung des Unterrichts nicht nur die Bedienung der Software – mit anderen Worten die Eingabe von Werten und Formeln – vermittelt werden kann, sondern gleichzeitig auch grundlegende Konzepte und Methoden wie etwa das IF;THEN;ELSE-Konstrukt zur fallabhängigen Wertebelegung von Zellen. Somit dient die Behandlung der Standardsoftware deutlich beiden oben dargelegten Zielen.

Des Weiteren lässt sich die Bedienung von Standard-Software meist (wenn auch nur eingeschränkt) auf Software anderer Hersteller übertragen – die Unterschiede zwischen Open Office und Microsoft Office sind zwar deutlich, nicht jedoch unüberwindbar groß. Ein Grundverständnis der Bedienung eines dieser Office-Pakete ermöglicht somit ebenfalls eine grundlegende Bedienung des Pendant.

Aus diesen Ausführungen können leicht einige Kriterien abgeleitet werden, die bei der Auswahl von Software für eine Verwendung im Informatik-Unterricht zugrundegelegt werden sollten. Sicherlich sind zahlreiche weitere Kriterien denkbar – folgende Auswahl soll jedoch aufgrund des Umfangs dieses Papers einmal genügen:

- Die Software sollte aktuell sein, d.h. keine Verwendung von Anwendungen, die 10 Jahre und älter sind. Zwar lassen sich grundlegende Konzepte auch hieran vermitteln, jedoch unterscheidet sich häufig bereits die GUI der Programme dermaßen von heutigen Versionen, dass von Standard-Software bereits nicht mehr gesprochen werden kann. Eine Übertragung der Kenntnisse auf moderne Software ist somit für Lernende schwierig.

- Der zur Verfügung stehende Funktionsumfang sollte eher gering als riesig sein. Zu großer Umfang einer Software wirkt leicht überwältigend (gerade für Anfänger). So ist z.B. zu überlegen, ob Microsoft Access für die Vermittlung von grundlegenden Datenbankfunktionen nicht "eine Nummer zu groß ist".
- Die Software sollte nicht zu "automatisiert" sein – was eine große Hilfe beim alltäglichen Gebrauch darstellt, erschwert oft das Erlernen grundlegender Funktionen. Als extremes Beispiel soll angeführt werden, dass mittels einer Software wie "Dreamweaver" sicherlich nur schwerlich HTML zu vermitteln ist – obwohl es sicherlich sehr begrüßenswert ist, im Alltag NICHT HTML-Code von Hand erzeugen zu müssen, sondern auf dermaßen automatisierte Software zurückgreifen zu können.
- Es sollte nach Möglichkeit eine deutsche Version der Software existieren und verwendet werden
- Wenn möglich, sollte die Software für den Schüler preisgünstig bzw. kostenlos zu erwerben sein oder zur "Standard-Ausrüstung" von Computern gehören – sicherlich wird bei vielen Lernenden der Wunsch bestehen, die Software auf dem Rechner "zu Hause" ebenfalls zu installieren (und sei es nur zum Üben...).

Aufgabe 5.)

Nur am Rande: Schöne Aufgabe! ;-)

Sehr geehrter Herr X Y,

haben Sie vielen Dank für die Mitteilung Ihrer Beobachtungen hinsichtlich der Divergenz zwischen Motivationslage von Schülerinnen und Schülern sowie den grundlegenden Forderungen des Faches Informatik (die sich meiner Meinung nach übrigens durchaus auf die Real- und Hauptschule übertragen lassen). Seien Sie sich zunächst gewiss, dass mir die von Ihnen beschriebene Problematik bekannt ist – ich jedoch keine Lösung nach Rezept präsentieren kann... noch existiert bzw. könnte eine solche existieren.

Die von Ihnen dargestellte Motivationslage von Lernenden beschreibt in meinen Augen den Wunsch nach Aufbau einer "Computer Literacy", d.h. einer "Beherrschung" des Computers. Dies erstreckt sich von der Bedienung von Standard-Software über die Nutzung des Internets bis hin zu einem grundlegenden technischen Verständnis der "Maschine" Computer. Kann bzw. soll das Schulfach Informatik dem nachkommen? – Eine Frage, die Sie mit "ja" beantworten.

Es liegt mir fern, Ihnen an dieser Stelle zu widersprechen – im Gegenteil! Jedwedes Schulfach ist nur dann lebendig, wenn es einen Bezug zur Lebenswelt der Lernenden aufweist und dazu gehört nicht zuletzt sicherlich auch ihre Motivationslage. Viel wichtiger, als die Frage, OB das Schulfach Informatik diesen Bedürfnissen nachkommen soll ist meines Erachtens daher, WIE dies geschieht. Soll bzw. muss zugunsten dessen die Vermittlung grundlegender Konzepte und Modelle der Informatik (so zum Beispiel das auch von Ihnen angeführte informatische Modellieren) leiden? Ich meine, nein!

Vielmehr ist es Aufgabe des Informatikunterrichts in der Sek.I, diese beiden Welten "in Einklang zu bringen", soll heißen, günstige Anknüpfungspunkte zu finden. Beispiele hierfür finden sich reichlich: So kann etwa bei der Behandlung von Standard-Software gleichzeitig eine Einführung in grundlegende Denkweisen der Informatik stattfinden – denkbar z.B. bei der Benutzung einer Datenbank zur Erstellung von Serienbriefen (gemeint sind hier beispielsweise "Abfragen", die die Struktur von

IF;THEN;ELSE-Anweisungen besitzen oder ähnliches). Ein anderer Ansatzpunkt wäre, das Unterrichtsthema "Erstellung von Homepages" mit einer Einführung in JavaScript zu kombinieren und in diesem Rahmen grundlegende(!) Konstrukte unterrichtlich zu behandeln.

Leider sind mir zur Zeit keine Werke bekannt, die das Spannungsfeld zwischen Lernermotivationen und Forderungen des Schulfaches Informatik explizit behandeln. Nach der Lektüre ihrer eMail plane ich jedoch, die Problematik den Studierenden in der von mir z.Z. an der Universität Dortmund durchgeführten Veranstaltung "Didaktik der Informatik für die Sek. I" vorzustellen und ihre Reaktionen im Rahmen einer Übungsaufgabe zu sammeln. Gerne werde ich Sie in diesen Prozess mit einbeziehen und Ihnen die entsprechenden Beiträge in anonymisierter Form zukommen lassen.

Mit freundlichen Grüßen,

L. Humbert

Aufgabe 1)

Allen in der E-Mail genannten Gründen gemeinsam ist, dass die Schülerinnen und Schüler die Bereitschaft zum Lernen ausdrücken – in dieser Hinsicht stellt sich allerdings die Frage, ob alle Schülerinnen und Schüler sich dort wiederfinden. Diese Gemeinsamkeit mag trivial wirken, ist aber Voraussetzung für Unterricht. Sieht man vom "Hilfsmittel Computer" ab, sind damit alle Berührungspunkte genannt.

Andererseits geht die Interessensschiere genauso weit auseinander wie die Anwendungsmöglichkeiten der Informatik. Angefangen bei (1) Hardwarefragen und Programmierung, über (2) klassische Anwendungssoftware, (3) Internet und multimedialer Einsatz, bis hin zu (4) Missbrauch ist alles vorhanden. Dabei gerät der erste Punkt deutlich ins Hintertreffen.

Aufgabe 2)

Die Informatik ist ein strukturelles Fach. Vor diesem Hintergrund gibt es nur eine Forderung: Alle sollen den Umgang mit Strukturen aus dem Bereich der Informatik erlernen. Dabei kann zwischen verschiedenen Kategorien des Umgangs und Lernens auf jeweils verschiedenen Stufen unterschieden werden. Die psychologischen und lerntheoretischen Modelle sind hinlänglich bekannt und ebenso, dass die informatischen Anwendungen sehr breit gefächert sind. Diese eine Forderung lässt sich also beliebig fein und beliebig nichts-sagend zerteilen.

Es gibt auch eine nicht informatik-relevante Begründung für den Umgang mit Strukturen: Ist dies einmal (zumindest teilweise) erlernt, kann das strukturelle Denken in anderen Bereichen desselben Faches angewendet oder in andere Fächer getragen werden und dort weiter helfen. Ohne eine strukturelle Basis ist dies nicht möglich. – Ein kleiner Einwurf dazu: Diese Behauptungen haben natürlich Konsequenzen für die anwendbaren Denk- und Lernmodelle. Falls es dazu Nachfragen gibt, können sie in der Übung geklärt werden.

Aufgabe 3)

Abgesehen von Desinteresse und keinen Erwartungen sind mögliche Erwartungen – ich bin schließlich weder Eltern, noch Schulleitung – seitens

- der Eltern: solide Grundbildung der Kinder – was auch immer das sein mag;
- der Schulleitung: Profilierung durch guten Informatikunterricht, guter Ruf der Schule (und der Leitung);
- der Schülerinnen und Schüler (Motivation – besser Beweggrund – aus Aufgabe 1 und Erwartung sind schließlich nicht gleichbedeutend): Arbeiten am Computer, weniger Kontrolle durch Lehrkräfte, mehr Freiheit;
- IHKs und künftige Arbeitgeber: Kenntnissvermittlung über die konkret eingesetzte (Standard-) Software;
- Sportvereine: Kenntnisse der Schülerinnen und Schüler über Tabellenkalkulationen für Ergebnististen;
- Microsoft^{ff}: Verwendung von Microsoft^{ff}-Produkten, insbesondere wenn sie preislich günstig oder kostenlos sind;
- SuSE: Verwendung freier (nicht kostenloser) Software im Informatik-Unterricht;

- SuSE PRESS: Verkauf von Büchern über SuSE-Linux an Lehrkräfte;
- Schulbuchverlage: Veröffentlichung von Büchern anerkannter Lehrkräfte;
- Politiker: . . . hier kann ich mir beim besten Willen nichts vorstellen.

Diese Auflistung ist natürlich nicht schwarz-weiss gemalt und wir werden uns auch nicht beugen – oder hab' ich die Aufgabe falsch verstanden? ;-)

Aufgabe 4)

Grundsätzlich gibt es keine Standardsoftware (Betriebssystem, Office-Paket, Internet- und Mail-Client), deren Einsatz ich ausschließen könnte. Neben Schultauglichkeit – dieser Begriff ist sicherlich genauso schwammig und ungeeignet wie die Tauglichkeitsprüfung für die Grundschule –, Anschaffungskosten und Folgekosten für Wartung und Schulung sind vorhandene Lizenzen ein wichtiges Kriterium.

Zusätzlich ist die Abstimmung mit dem übrigen Lehrpersonal von Bedeutung und, ob Standardsoftware überhaupt für einen konkreten Lehrgang geeignet ist.

Aufgabe 5)

Lieber Herr Y,

auch ich sitze zwischen den Stühlen, denn ich beantworte Ihre E-Mail an Herrn Humbert, ohne im Geringsten zu wissen, was er Ihnen schreiben würde.

[Schülergründe für Wahlpflichtfach Informatik]

Ich halte jeden dieser Punkte für den Informatikunterricht geeignet, obwohl ich einige eher in den Bereich einer AG ansiedeln würde und andere sicherlich nicht so behandelt sehen möchte, wie es die Schüler gerne hätten.

[Frage nach Konzepten, Material, etc.]

Im Bereich der Programmierung würde ich mich im Umfeld der Sprachen umsehen, die für Unterricht konzipiert wurden, etwa Comal 80, Logo, Modula-2 oder Pascal. Dort gibt es sicherlich Materialien.

Wenn es um Anwendungssoftware geht, würde ich bei der Konzeption eines Lehrganges auf Alternativen in der Wahl der konkreten Software achten. So gibt es sicherlich vier oder fünf gängige Office-Pakete; wird im Unterricht eines verwendet, so ist das Arbeiten mit einem anderen eine Art Gesellensstück. Dies ist sicherlich auch im Rahmen einer Fortbildung möglich.

Im Übrigen kann ich Sie nur bitten, mich über vorhandene oder selbst erarbeitete Konzepte auf dem Laufenden zu halten. Ich werde dies im Gegenzug ebenfalls tun.

*Beste Grüße,
XY*

Übung zur Vorlesung „Didaktik der Informatik III“

Arbeitsblatt 1:

Vor Bearbeitung dieses Aufgabenblattes möchte ich darauf hinweisen, dass sich die hier getroffenen Aussagen auf die im Skript zur Vorlesung vorgestellte E-Mail [1] beziehen und somit auch nur im Kontext dieser zu betrachten sind. Ein repräsentatives Bild kann allein auf Grund dieser E-Mail nicht erwartet werden, so dass in dieser E-Mail getroffenen Aussagen als noch zu überprüfen zu betrachten sind.

Aufgabe 1:

Die Motivationslage der Lernenden im Wahlpflichtfach Informatik der Sekundarstufe I stellt sich laut der oben genannten E-Mail so dar, dass durchaus ein reges Interesse an der Informatik besteht. Dabei werden verschiedenen Motivationsgründe angeführt, welche in verschiedene Kategorien einzuordnen sind.

- a) Motivation aus erwünschter Anwendungskompetenz:

(„ich brauche meinen Computer später für meine Ausbildung“; „ich will bessere Referate und Präsentationen machen“; „ich will eine Webseite machen“; „ich will lernen, wie man Emails und Briefe schreibt“; „ich will programmieren lernen“; „ich will CDs und Filme kopieren können“; „ich will mit dem PC Musik machen“; „ich will wissen wie man Musik und Filme aus dem Internet holen kann“ [2])

Diese Art der Motivation ist, allein aus der Quantität der Beiträge zu ersehen, wohl die entscheidende für Lernende in der Sekundarstufe I. Sie erwächst aus dem Wunsch das Informatiksystem als Werkzeug zum Erreichen verschiedener Ziele einsetzen zu können. Die Motivation geht somit nicht in direkter Weise vom Computer oder der Software aus, sondern vom zu erzielenden Effekt. Das Interesse der Lernenden gilt so z.B. der Tätigkeit des Hörens von Musik auf dem Rechner selbst, meist aber nicht den eigentlichen informatischen Inhalten.

- b) Motivation aus Interesse an Wirkprinzipien:

(„ich habe zwar einen Computer, kenne aber außer Spielen keine anderen Programme“; „ich will alles wissen, was ich brauche, um einen Computer zu kaufen“; „ich will meinen Computer selber reparieren können und nicht immer Freunde fragen“; 2)

Diese Motivation erwächst aus dem Wunsch der Lernenden nach mehr Transparenz bezüglich des Aufbaus und der Wirkprinzipien von Informatiksystemen. Im Gegensatz zu a) ist dabei nicht mehr allein die erfolgreiche Anwendung des Systems im Fokus des Interesses, sondern auch das Informatiksystem selbst.

- c) Motivation aus Unsicherheit:

(„ich will wissen, wie ich mich vor Dialern und Spam schützen kann“; [2])

Bei dieser Motivation spielen die Unsicherheit und auch die Sorge bezüglich eventueller Schäden eine besondere Rolle.

Aufgabe 2:

Die grundlegenden Forderungen des Faches Informatik bestehen in der Vermittlung folgender drei Bereiche: [3]

1) Grundwissen über Daten und Systeme

In diesem Bereich sollte die Verdeutlichung verschiedener Themenbereiche angestrebt werden:

- Grundlagen der Mensch-Maschine-Interaktion *
- Sprachen und Strukturen zur Informationsspeicherung, Wiedergewinnung und Verknüpfung
- Wirkprinzipien von Informatiksystemen *
- Telekommunikation, Grundbegriffe und Informationswege *

2) Wissen über Programme und Programmierbarkeit

Dazu gehören als Ziele:

- Berechenbarkeit in Programmen
- Grenzen der Programmierbarkeit
- Komplexität von Lösungen
- Verifikation von Lösungen

3) Exemplarisches Können zum Programmieren

Dabei sollen vor allem folgende Fertigkeiten vermittelt werden:

- Entwicklung von Informatiksystemen
- Inferenzprozesse und maschinelles Lernen

Dabei ist zu berücksichtigen, dass es sich hierbei um Anforderungen an die gesamte Informatikausbildung im Kontext einer allgemein bildenden Schule handelt. Unter Berücksichtigung der Idee eines Spiralcurriculums wäre es zwar wünschenswert auch in der Sekundarstufe I bereits auf die meisten der oben genannten Aspekte einzugehen, meiner Meinung nach sind aber manche in den zu betrachtenden Jahrgangsstufen besonders hervorzuheben. (Hervorzuheben sind diejenigen, welche mit einem * gekennzeichnet sind.) Diese finden sich in *Bereich 1*, dem „Grundwissen über Daten und Systeme“.

Aufgabe 3:

Die Erwartungen der Schulleitung sind in diesem Fall einfach zu benennen. Eine verantwortungsvolle allgemein bildende Einrichtung sollte es sich zur Aufgabe machen, eine zu vermittelnde Fachwissenschaft in angemessener Weise darzustellen, was zu den unter *Aufgabe 2* genannten Zielen führt. Darüber hinaus sollte den Lernenden ein Satz von praktischen Fähigkeiten im Umgang mit Informatiksystemen an die Hand gegeben werden, welcher einen gewissen Grundbedarf im späteren Berufsleben abdeckt und eine so gestaltete Grundlage bietet, dass die Lernenden in der Lage sind sich selbst neue erforderliche Inhalte anzueignen.

Dieser Grundstock von Fähigkeiten sollte auch das sein, was die Eltern von Lernenden von einer allgemein bildenden Schule erwarten.

Aufgabe 4:

Wie bereits in *Aufgabe 3* angesprochen, sollte eine Grundlage von Fertigkeiten vermittelt werden, auf denen von den Lernenden selbst aufgebaut werden kann. Mit der reinen Vermittlung von Arbeitstechniken mit Standartanwendungen ist diese jedoch nicht zu erreichen, da die eigentlichen Wirkprinzipien auf diese Weise im Verborgenen blieben. Daher sollte meiner Meinung nach bei der Vermittlung von allgemeinen Wirkprinzipien auf den Einsatz von Standardsoftware verzichtet werden.

Dennoch sollte eine Auseinandersetzung mit ihnen stattfinden, indem an geeigneter Stelle verdeutlicht wird, dass die erlernten Wirkprinzipien auch auf solche Softwareprodukte zutreffen und somit eine Anpassung an die Arbeit mit ihnen ohne größeren Aufwand möglich ist. Sie sollten jedoch nicht im Vordergrund einer schulischen Ausbildung stehen.

Aufgabe 5:

-

[1] vgl. Skript zur Vorlesung „Didaktik der Informatik für die Sekundarstufe I“, Kapitel „1.2 Sekundarstufe I“

[2] vgl. Präsentationsfolien zur Vorlesung „Didaktik der Informatik für die Sekundarstufe I“, 1. Vorlesung, S. 1

[3] vgl. Sigrid Schubert; Begleitmaterial zur Vorlesung: Einführung in die Didaktik der Informatik; Universität Dortmund; 2002; S. 67 ff.

1. Aufgabe

Beschreiben Sie die antizipierte Motivationslage der Schülerinnen eines Wahlpflichtfaches Informatik in der Sekundarstufe I in einer Real- oder Hauptschule.

Im Folgenden möchte ich versuchen, die vom Lehrer geschilderte (und damit natürlich durchaus nicht objektiv erfasste) Motivationslage der Schüler bei der Wahl eines abschlussrelevanten Faches Informatik zu gliedern, zu charakterisieren und bestimmten Oberpunkten zuzuordnen.

Der erste Oberpunkt, unter dem ich die Ansichten der Schüler festhalte lautet "Nutzung für Schule und Beruf". Hierunter fallen der Wunsch Kommunikationsmittel wie E-Mail und Briefe schreiben zu können, aber auch die etwas vage Aussage, dass man die Computer für die "eigene Ausbildung" benötige. Auch den Wunsch nach besseren Möglichkeiten für "Präsentationen und Referate" würde ich unter diese Kategorie fassen.

Der zweite Oberpunkt lautet "Technische Kompetenz". Hierbei geht es nicht mehr um die Nutzung des Informatiksystems mit einem bestimmten Ziel, sondern um eine allgemeine Kompetenz, die es ermöglicht Informatiksysteme zu erwerben, also die "Güte" zu beurteilen und auch zu reparieren, wobei nicht näher festgelegt ist, welcher Art diese Reparatur ist (vermutlich handelt es sich nicht um eine umfangreiche Hardware-Reparatur, sondern lediglich um die Neuinstallation einer Anwendungssoftware oder eines Betriebssystems). Der dritte Punkt ist mehr ein "allgemeines Interesse" an den Möglichkeiten des Rechners an sich. Diesem würde ich die Unterpunkte "eigene Website", "keine Programme außer Spiele" genauso zuordnen wie "programmieren lernen".

Der letzte und vermutlich aussagekräftigste Punkt lautet "Freizeitgestaltung". Hierunter fallen "Musik machen", "Filme kopieren" und "Musik und Filme aus dem Internet holen". Diesen Punkt unterscheidet sich bewusst vom ersten Punkt, da hierbei nicht so sehr die Kommunikation, sondern das Ziel im Mittelpunkt steht.

Mit Ausnahme des dritten Punktes, der allerdings auch die wenigsten Unterpunkte hat, sind all diese Probleme aus dem alltäglichen Leben mit einem sehr geringen Abstraktionsgrad. Die Schüler sind an dem Rechner in erster Linie als "Werkzeug" interessiert, um ein bestimmtes Ziel zu erreichen, allenfalls noch als Spielzeug, aber auch nur um an andere Medien zu gelangen gestalterische Wünsche wie "eigene Website" und "programmieren lernen" fallen in die weniger stark genutzte Kategorie.

Fazit: die Schüler nennen zumeist praktische und realistische Zielsetzungen, die in der Tat für sie - zumindest kurzfristig - einen enormen Nutzen nach sich ziehen. Da diesbezüglich keine Angaben vorliegen, lässt sich nur schwer erahnen, ob an einem Gymnasium die Bereiche ähnlich aussehen würden. Es steht zu vermuten, dass einige Schüler zumindest "hacken" oder "Netzwerke" genannt hätten. Ein direkter repräsentativer Vergleich solcher Vorstellungen wäre meines Erachtens interessant...

2. Aufgabe

Zeigen Sie auf, welche "grundsätzlichen Forderungen des Faches" bestehen.

Wie jedes Fach hat auch das Fach Informatik die Bestrebung möglichst langfristige Bildungsinhalte zu vermitteln. Außerdem sollen diese nicht in Form einer Produktschulung geschehen, sondern verallgemeinerbare (und damit auch abstraktere) Inhalte transportieren. Konkret schreiben die Lehrpläne vor: "Umgang mit Dateiverwaltungssystemen", "Nutzung von Textverarbeitungssystemen", Nutzung vernetz

ter Kommunikationssysteme", "binäre Informationsdarstellung", "elementare logische Schaltungen", "Messen, Steuern, Regeln", sowie "Problemabstraktion". All diese Ziele sind von eher spezieller Natur. Sie vermitteln ein relativ weitreichendes Wissen über den Aufbau und die Funktionsweise ein Rechners an sich, das vermutlich auch für einen längeren Zeitraum in dieser Form noch aktuell sein wird. Unmittelbar in dieser Form werden sie einem Schüler der Sekundarstufe I an einer Realschule höchstwahrscheinlich nur in wenigen Ausnahmefällen im späteren Berufsleben begegnen.

3. Aufgabe

Welche Erwartungen an den Informatikunterricht bestehen seitens der Eltern, der Schulleitung, . . . (bitte weitere Interessensgruppen ergänzen)?

Als Interessensgruppen wären die Eltern, die Schulleitung, die Wirtschaft, die Hochschulen, die allgemeine Didaktik, Pädagogik und Erziehungswissenschaften, die Politik, ausgebildete sowie nichtausgebildete Informatiklehrer/Studierende sowie Lehrer, die nicht Informatik unterrichten.

Von Seiten der Eltern werden vermutlich in erster Linie die praktischen Anwendungen dominieren. Ohne dabei repräsentative Angaben machen zu können, vermute ich, dass viele Eltern vermutlich Kenntnisse von MS Word, Access und Excel, also den wichtigsten Bestandteilen des Office-Pakets, fordern würden. Dazu kommen vermutlich einfache alltägliche Rechercheaufgaben im Internet wie die Abfahrtszeiten von öffentlichen Verkehrsmitteln festzustellen oder den Einsatz als Lexikaersatz.

Die Wirtschaft wird vermutlich hier divergentere Ansichten äußern. Einige Wirtschaftszweige werden vermutlich die Nutzung bestimmter Softwarepakete (also im wesentlichen des Office-Pakets) voraussetzen. Andere werden eher ein allgemeines Verständnis (z.B. im Bereich Netzwerke) bevorzugen.

Wie die Erwartungen der Hochschulen in diesem Bereich, der sie ja, da die Schüler im Allgemeinen kein Hochschulstudium anstreben, nur teilweise betrifft, aussehen würden, ist fraglich, da vermutlich wenig Verständnis für die gesellschaftliche Bedeutung existiert. Vermutlich werden viele fordern, dass der Unterricht wissenschaftlich korrekt und dementsprechend einen hohen Abstraktheitsgrad enthält. Neben vielen der "grundsätzlichen Forderungen des Faches" werden viele Informatiklehrer vermutlich zudem verlangen, dass keine Produktschulung stattfindet, dass langlebiges Wissen vermittelt wird, dass Verständnis für die prinzipiellen Grenzen der Rechner wie Grundprinzipien der Programmierung sowie bei einigen evtl. auch von einzelnen Programmiersprachen gefordert wird.

Außerdem erheben vermutlich viele Informatiklehrer auch den Anspruch, dass der Informatikunterricht auch für sie selbst interessant bleiben soll. Andere Lehrer/Interessensverbände von Lehrenden könnten evtl. eine Einstellung des Informatikunterrichts an sich verlangen, um ihre jeweiligen Fächer zu stärken. Dazu kommen Forderungen, die auf die Nutzung der Informatiksysteme in ihrem Unterricht abzielen, beispielsweise die Nutzung bestimmter dynamischer Geometriesoftware wie Euklid (vgl. www.dynageo.de) für den Mathematikunterricht und mit der Nutzung bestimmter Betriebssysteme verbunden sind. Diese Wünsche beeinflussen natürlich auch massiv den Informatikunterricht an sich. Auch die Schulleitung könnte eine solche Position beziehen.

Die Softwareindustrie hat natürlich das Interesse, dass bereits im Informatikunterricht ihre Softwareprodukte eingesetzt werden; ähnliches gilt für die Hardwareindustrie (beispielsweise gibt es die Möglichkeit im Unterricht durch Steuern und Regeln Ampelanlagen herzustellen, hierzu sind zusätzliche Hardwarekomponenten notwendig, die natürlich angeschafft werden müssen)

Nicht zuletzt bestehen auch von Seiten der "allgemeinen Didaktik" (Erziehungswissenschaften wie Pädagogik) evtl. Ansprüche im Bezug auf eine pädagogische Eignung bestimmter Informatiksysteme z.B. im Hinblick auf die bestehenden Lerntheorien und auf die Bedeutung des Informatikunterrichts im Bildungskontext.

Natürlich haben auch Bildungspolitiker Ansprüche. Vermutlich würden sie sich für einen zeitgemässigen Informatikunterricht mit hoch modernen Informatiksystemen einsetzen (wobei die Frage der Finanzierung offen bleibt).

4. Aufgabe

Wie schätzen Sie den Einsatz von "Standardsoftware" ein? Geben Sie ggf. an, auf welche Kriterien sie bei ihrer Auswahl besonderen Wert legen oder andernfalls, warum sie sich gegen den Einsatz entscheiden.

Die Problematik hierbei liegt bei der Definition des Begriffs "Standardsoftware". Es kann die Festlegung eines bestimmten Standards für alle Schulen gemeint sein, aber auch die Verwendung bereits bestehender Standards in der Wirtschaft oder bei privaten Nutzern. Sollten Standards für die Schulen gewählt werden, die denen der Wirtschaft (auch Eltern, Ausbildungsbetriebe, etc..) widersprechen, ist mit massiven Widerständen von Seiten der Eltern zu rechnen, die meistens dazu tendieren, die Ziele der Schul- und Informatikausbildung eher kurzfristig und auch sehr konkret in bestimmter Anwendungssoftware zu sehen. Generell besteht bei Standards im Bereich der Software immer das Problem, dass Standards im Allgemeinen nicht dazu neigen, über längere Zeiträume konstant zu bleiben, d.h. eine Ausbildung in bestehender "Standardsoftware" birgt grundsätzlich das Risiko kurzfristige Erfolge zu erzielen, die bei Änderung des Standards nicht mehr vorhanden sind. Andererseits hat auch die Wahl von "Nicht-Standards" Konsequenzen. Standards garantieren die Vergleichbarkeit von Fähigkeiten, Fertigkeiten und Wissen auch gegenüber nicht fachkompetenten Personen und damit eine Beurteilung und Bewertung nicht nur in der Schule, sondern auch in späteren Ausbildungs- und Einstellungssituationen. Andererseits ist es durch die zu starke Betonung von Standards aller Wahrscheinlichkeit nicht möglich wirkliches Verständnis für einen Themenbereich an sich zu erfassen (wer beispielsweise WordTM beherrscht, sollte eigentlich in der Lage dies auf andere Textverarbeitungssoftware zu übertragen; die Frage ist lediglich, ob man dies einfach so voraussetzen kann und sich daher alternative Software nie ansieht).

In gewisser Weise ist dieser Widerspruch zwischen Aktuellem, konkret anwendbaren Fähigkeiten und langfristig einsetzbaren Ideen eine Version des Widerspruchs zwischen formaler (also abstrakter) und materialer (also inhaltlicher) Bildung. Da die Wahl keines der beiden Extrema eine Lösung des Problems darstellt, kann eine Lösung nur darin bestehen sich zumindest teilweise auf Standards einzulassen, zumal diese ja eine Vergleichbarkeit des Unterrichts und auch eine Offenlegung von Zielvorstellungen an Nicht-Experten darstellen. Hierbei ist es m.E. allerdings sinnvoll, diese Standardisierung so gering wie nur irgend möglich zu halten. Dies könnte man dadurch umsetzen, dass man die Standardkomponenten im Unterricht stark eingrenzt und häufiger darauf verweist, dass es auch noch andere Systeme gibt.

5. Aufgabe

Beantworten Sie die E-Mail.

Sehr geehrter Herr Y,

Ich möchte mich bei Ihnen sehr für diese E-Mail bedanken, die meiner Ansicht nach das Grundproblem, das derzeit in der Sekundarstufe I in der Informatik besteht, sehr genau trifft: den Widerspruch zwischen den Anforderungen einer formalen Bildung und denen einer materialen Bildung, die für die Schüler von konkreten Nutzen in ihrem späteren Berufsleben ist.

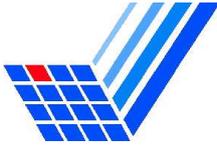
Um ihre wichtigste Frage zu beantworten: Mir als Student des Lehramtes Informatik sind momentan nur wenige Konzepte bekannt, die eine direkte Kombination mit Elementen erlauben, die die Schüler im Unterricht motiviert.

Denkbar wäre es, die Entwürfe Peter Hubwiesers (vgl. Hubwieser, P. (2000). Didaktik der Informatik Grundlagen, Konzepte, Beispiele. Springer) bzgl. der Sekundarstufe I anzupassen. Dies wäre allerdings eine unbefriedigende Lösung, da dies nicht den Kern der Motivation Ihrer Schüler trifft und zudem fraglich ist, ob der Abstraktionsgrad angemessen ist. Ich persönlich stimme dem Einsatz von Standardsoftware im Schulunterricht auch nur soweit zu, als dass den Anforderungen des Faches ebenfalls Rechnung getragen werden kann. Mit anderen Worten: nur eine Kombination beider Zielsetzungen: der langfristigen, formalen Lösung durch die "Forderungen des Faches" und der "konkreten Anforderungen des Lebens" können einen sinnvollen Informatikunterricht garantieren. Prinzipiell sehe ich daher für dieses Problem zwei Möglichkeiten. Man könnte den Informatikunterricht zweiteilen und an mehreren Stellen gezielt die Kernelemente der "Inhalte für das Leben" unterbringen. Dies hätte natürlich zu Konsequenz, dass der Informatikunterricht schubweise verläuft und dass eine Absicherung des Lehrers durch die Richtlinien und Lehrpläne nicht unbedingt gegeben ist. Eine weitere Möglichkeit wäre es, zu versuchen die "konkreten Elemente" mit den geforderten Inhalten zu verbinden. Dies erfordert vom Lehrer eine grosse fachliche Kompetenz und zudem didaktisches Geschick. Konkrete Umsetzungen dieser Methode müssten also von den Lehrern selbst erbracht werden, was eine schwierige Aufgabe ist. Alles in allem: ebenfalls keine befriedigende Lösung . . .

Obwohl ich Ihnen bei ihrem konkreten Problem vermutlich wenig hilfreich war, möchte ich mich trotzdem bei Ihnen für diesen wertvollen Hinweis bedanken, der mich als Sek II-Studierenden erstmals auf diese Problematik aufmerksam gemacht hat. Möglicherweise ist es Ihnen möglich, Kontakt zu halten und so Ihre Erfahrungen im Rahmen der Lehrerfortbildung auch anderen zugänglich zu machen?

Mit freundlichen Grüßen,

XY



Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 2

Ein insbesondere in Österreich weit verbreiteter Ansatz zur informatischen Bildung besteht in der Zertifizierung durch den Europäischen Computer Führerschein™ (ECDL)¹. In dieser Übung soll geprüft werden, inwieweit sich Aspekte dieses Konzeptes auf den Informatikunterricht in der Sekundarstufe I übertragen lassen (können).

1. Aufgabe: (3 Punkte)

Beschreiben Sie den ECDL, indem Sie folgende Fragen beantworten:
Wie ist er entstanden? Welche Lernziele bzw. Prüfungsinhalte umfasst er? Welche Ziel-
dimensionen vermissen Sie?

2. Aufgabe: (4 Punkte)

Im dem Modul 1 (Grundlagen der Informationstechnologie – IT) des ECDL wird eine Auswahl von 8 Fragen eines Fragenkatalogs² getroffen, die in der Prüfung im Freitext beantwortet werden sollen. Um den Korrekturaufwand zu verringern, können mögliche Antworten für ein Multiple-Choice-Verfahren vorgegeben werden. Wählen Sie sich für einen Multiple-Choice-Test vier Fragen aus, für die Sie jeweils vier Antworten erstellen (2 Punkte). Geben Sie zusammenfassend die Vor- und Nachteile sowie die Einsatzschwerpunkte von Freitext- resp. Multiple-Choice-Aufgaben an.

3. Aufgabe: (3 Punkte)

Betrachten Sie die eine Auswahl von Beispielprüfungen³ des ECDL. Welchem Anforderungsniveau entsprechen die Prüfungsfragen? Können Sie sich vorstellen, dass Schülerinnen⁴ ihrer zukünftigen Schule zum Ende der Sekundarstufe I diese Prüfung bestehen (sollten)? Welche Bereiche würden Sie reduzieren (oder entfernen), welche erweitern (oder hinzufügen)? Begründen Sie Ihre Entscheidungen.

Abgabe: 29.10.2003

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ <http://www.ecdl.at>

² <http://www.gym1.at/schulinformatik/aus-fortbildung/fachdidaktik/kv-01/grundlagenfragen.pdf>

³ http://www.ecdl.at/syllabus/beispiel/Beispiel-Pruefungen/Beispiel_Pruefung.pdf

⁴ Auch auf diesen Übungsblättern wird das generische Femininum verwendet. Die männliche Schreibweise ist implizit mit gemeint.

1. Aufgabe:

Der Europäische Computerführerschein (ECDL) ist ein international gültiges Zertifikat, mit dem die grundlegenden Fähigkeiten und Fertigkeit mit einem Informatiksystem im Bereich Büroanwendungen nachgewiesen werden können. Ziel ist die Standardisierung der grundlegenden Fähigkeiten für Wirtschaft, Verwaltung und Schule im Umgang mit dem Rechner. Verantwortlich sind die Comptergesellschaften Europas sowie ihre Dachorganisation CEPIS. Der Test setzt sich aus folgenden Modulen zusammen:

1. Grundlagen der Informationstechnologie
2. Computernutzung und Dateimanagement
3. Textverarbeitung
4. Tabellenkalkulation
5. Datenbank
6. Präsentation
7. Information und Kommunikation

Der Test ist unabhängig von einzelnen Softwareherstellern und kostet insgesamt 26,50 € pro Modul, zuzüglich 47 € für die Skill-Card, insgesamt also 233,- €. Der Führerschein ist unbegrenzt gültig, jedoch dürfen zwischen dem Bestehen der ersten und der letzten Teilprüfung insgesamt höchstens drei Jahre vergehen. Hierbei können bei Nichtbestehen einzelne Teile des Testes wiederholt werden. Historisch gesehen stand der ECDL, nachdem erstmals im Jahre 1994 die finnische Computergesellschaft die Bedürfnisse der Wirtschaft erhoben hat und die bis heute gültige siebenteilige Modulstruktur erarbeitet hat. Ein Jahr später griff der europäische Verband (CEPIS) diese Struktur auf und entwickelte sie in einer Arbeitsgruppe wieder. Diese wurde schließlich in eine unabhängige Organisation umbenannt, die ECDL-Foundation. In den folgenden Jahren führte eine Vielzahl europäischer Länder den ECDL ein. Lizenznehmer sind derzeit Belgien, Dänemark, Deutschland, Estland, Finnland, Frankreich, Griechenland, Großbritannien, Irland, Italien, Lettland, Litauen, Niederlande, Norwegen, Portugal, Rumänien, Russland, Österreich, Schweden, die Schweiz, Slowenien, Spanien, Tschechien, Ungarn und Zypern. Schließlich wurde im Jahre 2001 eine neue Version des ECDL der so genannte ECDL-ADVANCED eingeführt. Dieser kann für vier Module erworben werden: AM3-Fortgeschrittene Textverarbeitung, AM4-Fortgeschrittene Tabellenkalkulation, AM5-Fortgeschrittene Datenbank, AM6-Fortgeschrittene Präsentation.

Das erste Modul umfasst die theoretischen Grundlagen: Dies sind in erster Linie Wissen über Hard-, Software und IT, sowie die Hauptbestandteile eines Rechners, die Hardware, Software, ein grobes Verständnis von Netzwerken sowie des Internets, Grundkenntnisse über Computer am Arbeitsplatz, Ergonomie, Datensicherheit sowie das Urheberrecht. Es handelt sich also um Wissen, also um die erste Stufe der kognitiven Lernziele.

Strukturell lassen sich die angegebenen Grundlagen unterteilen in Wissen über Hardware, Wissen über Software, Wissen über Netzwerke sowie Wissen über Datensicherheit und Wissen über das Urheberrecht. Interessant ist insbesondere die letzte Komponente, da sie das Gebiet der Informatik ja fast verlässt und an der Grenze Informatik-Jura angesiedelt ist (nichts desto trotz aber von nicht unerheblicher Bedeutung ist).

Im zweite Modul werden die Grundlagen der Computerbenutzung gelehrt. Dies umfasst im Wesentlichen die Nutzung des Rechners an sich (vom Einschalten bis zu der Erstellung einer einfachen Datei), den Umgang mit einer graphischen Oberfläche, Dateiverwaltung, die Nutzung eines Virenschanners sowie das Drucken von Dokumenten. Anders als im ersten Modul umfassen dieses also neben dem "Wissen" und "Kennen" auch die Anwendung und das Verständnis, dazu kommen psychomotorische Lernziele wie "Imitation", "Manipulation" und auch in erheblichem Maße "Präzision" (bei der Verwendung eines Eingabegerätes wie der Maus).

Im dritten Modul "Textverarbeitung" geht es um die Benutzung eines der gängigen Textverarbeitungsprogramme, der gängigen Schritte wie Öffnen, Schließen, Formatierung, von Objekte, der Erstellung

von Serienbriefen sowie der Ausgabekontrolle. An dieser Stelle fällt auf, dass nun auch die Lernzielbe-
reiche Analyse (bei der Erstellung von Serienbriefen, muss die Ausgangssituation analysiert werden.) In
gewissem Maße müssen die Handlungen auch gegliedert werden, was in den psychomotorischen Bereich
fällt (dies ist bei der Erstellung eines einfachen Datei erforderlich). Interessant ist, dass in den Angaben
hier von Objekten die Regel ist, allerdings nur der Umgang mit Objekten im Sinne der Informatik ge-
meint ist. Obwohl hier natürlich Objekte manipuliert und eingesetzt
werden, handelt es sich hier m.E. noch nicht um Objektorientiertheit, da der praktische Umgang und die
Nutzung mit Objekten wie Grafiken oder Tabellen allein ohne die Begrifflichkeit der objektorientierten
Modellierung wie "Vererbung", "Methode" und "Attribut" und deren Zuordnung, mehr oder weniger all-
täglich ist.

Das Modul "Tabellenkalkulation" besteht aus der Nutzung des Programmes, dem Umgang mit Zeilen,
Spalten und Zellen sowie der Einfügung von Arbeitsblättern und Diagrammen sowie der Verwendung
verschiedener Formatierungen. Meiner Ansicht nach handelt es sich hierbei um die gleichen Lernzielbe-
reiche wie oben unter Textverarbeitung. Im kognitiven Bereich wird m.E. die Stufe der Synthese noch
nicht erreicht, da es sich eher um Anwendungen von bekannten Konzepten (wie beispielsweise der Sum-
mierung bei Serienbriefen) geht und nicht um die Erstellung oder fundamentale Neukombination von
Konzepten. Im nächsten Modul geht es um die Benutzung von Datenbanken (also um Anwendung), die
Erstellung von Formularen, das Abfragen von Information, das Erstellen von Berichten und des Vorbe-
reitens des Ausdrucks. Dies sind ebenfalls Lernziele aus den oben angesprochenen Bereichen. Im Mittel-
punkt steht die Anwendung, das Beachten und Reagieren sowie im psychomotorischen Bereich (die Prä-
zision und Handlungsgliederung).

Im sechsten Modul sind die Lernziele wiederum zunächst das Starten und Erstellen einer Präsentation
(also Beachten, Manipulation und Anwendung) genutzt, es folgt das Einfügen von Bilder, Diagrammen
und Effekten sowie wiederum der Ausdruck.

Im siebten und letzten Modul werden die Grundkonzepte des Internets gelehrt, also im Wesentlichen die
Kenntnis von Fachbegriffen, die Navigation im Netz und die Nutzung von E-Mail-Programmen.

Fazit: Im Wesentlichen stammen die Lernziele aus den Stufen "Kenntnis" und "Anwendung" und
wesentlich seltener "Verständnis". "Analyse" findet vor allem im Bereich "Serienbriefe" und
"Tabellenkalkulation" Anwendung. Bei den affektiven Lernzielen werden nur "Aufmerksam werden",
"Beachten" und "Reagieren" verwendet.

Es wäre m.E. sinnvoll das "Werten" stärker zu betonen. Relativ weit reichen die Ansprüche im psycho-
motorischen Bereich relativ weit. Zur Erstellung einer Präsentation müssen die Handlungen beispiele-
weise deutlich gegliedert werden, auch der Bereich der Naturalisierung wird angesprochen werden, die
Stufen stammen also in erster Linie aus allen Stufen des psychomotorischen Bereichs sowie erstaunlich
vielen Stufen des kognitiven Bereichs. Interessanterweise wird der affektive Bereich nur in sehr
geringem Maße angesprochen (die höchste erreichte Stufe der Taxonomie ist "Reagieren").

2. Aufgabe:

1. Was sind Computerviren?

- Programmstücke, die sich unerwünscht an andere Programme anhängen und Fehlfunktionen hervor-
rufen
- Programme mit falschem Namen
- Programme, die unerwünschte Werbemail verschicken
- Programme, die die Umstellung auf das Jahr 2000 nicht vollziehen könnten

2. Welches ist kein Dienst im Internet?

- TELNET
- FTP
- GOOGLE

- E-Mail

3. Sie haben eine große Menge von Information (Text und Bilder) schriftlich vor sich und möchten sie mit Hilfe eines Computers bearbeiten. Was benötigen Sie, damit Sie den Text nicht eingeben müssen?

- Scanner
- Microsoft TEXTMAGIC™
- Boxen
- Graphikkarte mit TV-Out

4. Wobei ist ein Mensch besser als ein Computer?

- Sortieren
- Vergleichen von Zeichenfolgen
- Spracherkennung
- Suchen in großen Datenmengen

Multiply-Choice-Aufgaben schränken die Antworten im Allgemeinen drastisch ein. Damit sind die deutlich einfacher zu korrigieren als Freitextaufgaben und ermöglichen es dem Lernenden zudem die Antwort zu erschließen, was zu einem recht hohen Beliebtheitsgrad führt, da die Aufgaben als einfacher gelten. (Dies ist Vor- wie auch Nachteil) In dem vorliegenden Multiple-Choice-Test lassen sich eine ganze Reihe von Fragen auch schon durch den "gesunden Menschenverstand" erschließen, was m.E. ein großer Nachteil ist. (Beispiel: Welches der folgenden ist ein Eingabegerät? Bildschirm, Drucker, Tastatur, Lautsprecher). Zudem sind Multiple-Choice Aufgaben relativ einfach neu zusammen zu stellen und zu rekombinieren (während die Erstellung nicht trivialer Aufgaben mit großem Aufwand verbunden ist) und die Antwort steht meist (scheinbar) objektiv fest, "unfares" Verhalten von Seiten des Kontrollierenden ist somit unwahrscheinlicher. Allerdings sind auch Missverständnisse in Bezug auf die Aufgabenstellung möglich, die in einem Freitext offenkundiger und entsprechend leichter zu bewerten wären, da der Gedankengang des Prüflings hierbei klarer ist. Durch Variation des Antwortteils ist zudem der Schwierigkeitsgrad variabel. Ein großer Nachteil ist, dass sich bei ungeschickt verfassten Multiple-Choice-Fragen und geringer Itemanzahl die Lösung unter einigen Annahmen bezüglich der Struktur der Aufgaben auch mit minimalen fachlichen Kenntnissen erraten lässt. ("Einmal muss doch 'nein' vorkommen"). Zudem verführen Multiple-Choice Aufgaben bei komplexeren Aufgabenstellungen dazu, Antworten ohne ausführliches Nachdenken anzukreuzen, da der Aufwand der äußeren Handlung (des Ankreuzens) und der Aufwand der geforderten Leistung (das Nachdenken, evtl. das Lösung einer Rechnung) stark divergieren.

Der größte Nachteil von Multiple-Choice-Aufgaben besteht darin, dass weiter gehende Überlegungen, die auch über die Vorüberlegungen des Lehrenden hinausgehen können, unterbunden werden und die Gedankengänge der Lernenden genormt werden, sowie in der Tatsache, dass der Lösungsweg resp. Gedankengang nicht niedergeschrieben werden und damit auch nicht Teil der Bewertung sind.

Angesichts dieser Vor- und Nachteile sollte man den Einsatzschwerpunkt von Multiple-Choice-Aufgaben vor Allem bei der Beantwortung relativ einfacher Sachaufgaben sehen, in denen im Wesentlichen Wissen abgefragt wird, das zudem einfach strukturiert sein muss. M.E. sollten hierbei nur wenige und auch nur triviale Schlussfolgerungen nötig sein, um die Frage beantworten zu können.

Generell halte ich persönlich den Einsatz von Multiple-Choice-Aufgaben aufgrund der oben genannten Faktoren für dermaßen problematisch, dass eine Anwendung auf wenige Fälle begrenzt werden sollte. Der Einsatz von Freitextaufgaben hat natürlich auch zahlreiche Nachteile. Im Wesentlichen sind die Gefahr abzuschweifen und themenfremde Aufsätze zu erstellen zu nennen. Nachteile liegen ferner in dem größeren Korrekturaufwand, sowie evtl. größeren Zeitaufwand. Die Vorteile treten vor Allem bei divergierenden Aufgaben zu Tage, bei denen sich der Lernende eine eigene Ansicht bilden soll (also insbesondere Bewertungsaufgaben) sowie Aufgaben, die mit hohen affektiven Lernzielen verbunden sind.

3. Aufgabe:

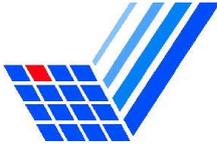
Meiner Ansicht nach könnte ein ähnlicher Test durchaus durchgeführt werden, allerdings wären zahlreiche Änderungen notwendig. Die meisten Aufgaben aus dem ersten Modul halte ich beispielsweise für obsolet, da die meisten Antworten durch logisches Denken ohne größere Probleme erschlossen werden können und zudem viele Begriffe mittlerweile in den allgemeinen Wortschatz übergegangen sind. Zudem wäre es sinnvoll den Test in mehrere Bereiche zu splitten, also den der Benutzung und des Dateimanagement schon in einer jüngeren Klasse durchzuführen. Der Bereich des Internets und der Kommunikation via E-Mail wäre hierbei deutlich vorzuziehen und direkt nach dem über Dateimanagement durchzuführen, da solche Kommunikationsmittel mittlerweile in vielen Bereichen eine ähnliche Bedeutung wie Telefon oder Fax haben. Auch der Teil über Rechercheaufgaben im Internet wäre deutlich zu vergrößern. Hinzu kämen – gerade in Folge der Novellierung des Urheberrechts – ein Exkurs über diesen Bereich sowie die Veröffentlichung von Websites auch unter diesem Aspekt. Den Bereich Präsentation würde ich kürzen, da viele dieser Inhalte auch mit den Kenntnissen über ein Textverarbeitungssystem herleitbar sind.

Ausserdem halte ich den Nutzen für Schüler der Sekundarstufe I in Haupt- und Realschulen für fraglich, da das Berufsbild m.E. derzeit nicht unbedingt diese Fähigkeiten verlangt.

Im Bereich der Anwendungen, die ungefähr die Bereiche in der Lernzieltaxonomie annehmen, würde ich stattdessen folgende Inhalte hinzufügen: Im Hinblick auf die Nutzung von Informatiksystemen im privaten Bereich halte ich die Installation und Deinstallation von Software für notwendig, da nur so eine unabhängige Tätigkeit im Rahmen einer privaten Nutzung des Rechners gewährleistet sein kann. Dazu kommen typische Wartungsaufgaben im Bereich der Software, wie beispielsweise das Durchführen von (bei einigen Betriebssystemen unerlässlichen Softwareupdates) sowie der De- und Reinstallation eines Betriebssystems, das bei bestimmten Produkten angesichts der geringen Halbwertszeit eines fehlerfreien Ablaufs unvermeidbar ist. Auch die Fehlersuche nach einer nicht-korrekten Installation wäre hier sinnvoll, allerdings auch im Rahmen der Sekundarstufe I schwer zu formalisieren. Notwendig ist auch die Kenntnis einzelner Programme (Office-Paket, StarOffice), Sprachen (LaTeX, Java) zumindest dem Namen und des Inhalts nach. Auch das Durchführen von einfachen Wartungsaufgaben im Bereich des PCs (beispielsweise aus Ein- und Ausbauen von Karten, das Anschließen eines Peripheriegeräts) halte ich für notwendig. Meiner Ansicht fehlen zudem an dieser Stelle viele Bereiche, die ich im Sinne einer Allgemeinbildung für Unerlässlich halten würde:

Dazu gehören vor Allem die ethischen Grenzen der Informatik, die bei solchen Lehrgängen häufig m.E. zu wenig Bedeutung finden, insbesondere die Punkte "Daten- und Persönlichkeitsschutz in einer vernetzten Gesellschaft" halte ich für besonders relevant. Hier kämen auch die Stufen "Beurteilen", "Werten" stärker als bisher hinzu.

Auch halte ich es für relevant Modellvorstellungen und ihre Grenzen und Kennen, Herzustellen und auch in ihrer Qualität bewerten zu können. Inhaltlich wäre eine Orientierung an den Konzepten aus (Hubwieser, P. (2001): Didaktik der Informatik, Springer) oder (Czischke, J. (1999): Von Stiften und Mäusen, Verlage für Schule und Weiterbildung) denkbar. Die Darstellung von Vektor- und Pixelgraphik und der objektorientierte Einstieg über die Vektorgraphik ermöglichen zudem eine Verknüpfung von Anwendungen, die im Kontext der beruflichen Weiterbildung relevant ist, sowie der objektorientierten Modellierung, die für die Fachdidaktik und die Fachwissenschaft von Bedeutung ist. Abgesehen davon halte ich den Einstieg in eine einfache Programmiersprache für geboten, da den Schülern nur so ein halbwegs korrektes Bild der Informatik vermittelt werden kann und die Schüler ihre eigenen Fähigkeiten und deren Grenzen besser einschätzen und auch erweitern können.



Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 3

In dem kollaborativen Arbeitsbereich zur Lehrveranstaltung (BSCW) wurden für Sie sechs Kapitel zur „Objektorientierten Modellierung von Textdokumenten“ bereitgestellt, die nach Ansicht der Autorin Siglinde Voß lehrplankonform im Fach Textverarbeitung der Realschulen in Bayern (Jahrgangsstufen 7 bis 10) genutzt werden können.¹ Lesen Sie für die Bearbeitung dieses Aufgabenblattes das Kapitel „06Seriendokumente“.

1. Aufgabe:

(3 Punkte)

Stellen Sie Gemeinsamkeiten und Unterschiede zwischen Seriendokumenten im WordXP- und im StarOffice6-Format heraus, in dem Sie die Struktur der Formate an Hand ihrer Klassendiagramme vergleichen. Geben Sie darüber hinaus an, welche weiteren Unterschiede in der Benutzung der Formate vorhanden sind. Resultieren diese Unterschiede aus der Verschiedenheit der Datenstrukturen?

2. Aufgabe:

(2 Punkte)

Geben Sie an, inwieweit ein Verständnis der zu Grunde liegenden Struktur von Seriendokumenten für die Benutzung von Textverarbeitungssystemen förderlich ist.

3. Aufgabe:

(3 Punkte)

In der Vorlesung und mit Übungsblatt 2 haben Sie sich mit dem ECDL-Konzept auseinandergesetzt. Zeigen Sie Unterschiede und Gemeinsamkeiten zum ECDL-Konzept auf.

4. Aufgabe:

(2 Punkte)

Beurteilen Sie die beiden Konzepte.

Abgabe: 6.11.2003

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ Dieses Material dürfen Sie unter keinen Umständen in digitaler Form weitergeben, da sich das Konzept noch in der Erprobung befindet.

1. Aufgabe:

Ein wesentlicher Unterschied in den Klassendiagrammen besteht darin, dass bei StarOffice6 das Datenquellendokument von einem anderen Dokumenttyp erbt als bei WordXP. Dies hat zur Konsequenz, dass es in Word theoretisch möglich wäre, dass ein Dokument gleichzeitig Datenquelle und Seriendokument bzw. Musterdokument sein kann (natürlich nur, wenn das Musterdokument den entsprechenden Format, also entweder eine Tabelle oder einen entsprechenden String mit den richtigen Trennzeichen). So wäre es möglich, dass zunächst Seriendokumente erstellt werden und dann als Datenquellendokumente verwendet werden. Dies ist meines Erachtens zwar praktisch von geringer Bedeutung aber nichtsdestrotz eine Funktion, die sich mit StarOffice6 nicht ohne Weiteres realisieren lässt, da hier die Datenquelle als Unicode-Textdokument (*.txt) mit einer logischen Tabelle oder gleich als Tabellenkalkulationsdokument gespeichert wird. Beide Formen enthalten also ein Musterdokument und ein Datenkontrolldokument, wobei allerdings bei StarOffice das Datenquellendokument von einer anderen Klasse, der Klasse Unicode-Textdokument, erbt. Die Struktur des Datenquellendokuments ist intern prinzipiell gleich. Beide bestehen aus genau einer Datentabelle, die beliebig viele Datenspalten und beliebig viele Datensätze enthält, wobei ein Datensatz je mindestens ein Datum enthält, das sich außerdem einer Datenspalte zuordnen lässt. Hierbei ist es insbesondere (lt. Klassendiagramm) explizit möglich, dass ein Datensatz leer ist. Hieraus folgt jedoch unmittelbar, dass auch die (in jedem Fall vorhandene) Datenspalte leer sein muss.

Der einzige Unterschied in der Struktur der Datenquellendokumente besteht darin, dass bei StarOffice6 bei einem Datensatz als Attribut vermerkt wird, ob dieser für irgendein Dokument ausgewählt wurde. In Word wird dies im Musterdokument vermerkt. Dies hat als praktische Konsequenz, dass unter Windows ausgewählte Datensätze bei Wiederverwendung des Datenquellendokumentes neu ausgewählt werden müssen, während bei StarOffice6 der Datensatz immer ausgewählt bleibt, auch wenn das Musterdokument ausgetauscht wird.

Ein fundamentaler Unterschied, der aus den Klassendiagrammen erkennbar ist, besteht darin, dass unter StarOffice Musterdokumente und Seriendokumente vom gleichen Typ sind, während unter Word hier eine andere Klasse - das Seriendokument - vorliegt. Unter StarOffice ist es demzufolge möglich eine tiefgehende unter Umständen weitverzweigte Hierarchie aus Musterdokumenten aufzubauen, bei denen jeweils nur Teile der Datenplatzhalter ersetzt werden. Dies ist möglich, da Datenplatzhalter aus verschiedenen Datenquellendokumenten einsetzbar sind. Ist ein Datenplatzhalter nicht in der Datenquelle ausgewählt, so wird er einfach kopiert.

Ein anderer, aber bei weitem nicht so relevanter Unterschied liegt darin, dass das Word-Dokument aus Abschnitten besteht, die wiederum aus Absätzen zusammengesetzt sind, die ihrerseits wiederum aus Zeichen zusammengesetzt sind. Ferner enthält das Dokument in beiden Produkten mehrere Datenplatzhalter, denen jeweils eine Datenspalte zugeordnet ist. Im übrigen liegen - dies ist jedoch unabhängig von den Klassendiagrammen - bei StarOffice die Datenquellendokumente in Datenquellendatenbanken vor, die bei StarOffice angemeldet werden müssen. Die Datenquellenbank enthält wiederum die Datenurl (es ist also auch eine entfernte Datenquelle möglich, was ein interessanter Aspekt im Bezug auf Netzwerke ist) und den Datentyp. An dieser Stelle ist die vorliegende Dokumentation m.E. etwas unklar, ob der Datentyp ausschließlich in der Datenquellenbank gespeichert wird und zu welchem Zweck. Es könnte hier eine Gefahr für inkonsistente Daten bestehen. Interessant ist auch, dass die Datenquellenbank als Ordner vorliegt, also prinzipiell auch außerhalb von StarOffice editierbar ist (dies entspricht dem allgemeinen Linux-Konzept).

2. Aufgabe:

Ein grundsätzliches Verständnis liefert dem Nutzer zunächst ein generelles Verständnis der zugrundeliegenden Strukturen. Vielen Word-Nutzern ist momentan nicht bekannt, dass es mehrere Formate gibt und dass ein Word-Dokument, als Unicode-Textdokument angezeigt ein grundsätzlich anderes Aussehen hat als im Word-Editor. Ein tieferes Verständnis liefert somit die Möglichkeit, dass Nutzer die

einzelnen Dateiformate besser nutzen können und so auch Anwendungsfehler wie den Verlust einer Formatierung bei Speicherung als Unicode-Dokument verstehen und nicht nur vermeiden lernen. Ein weiterer Punkt ist das Lernen der grundsätzlichen Begriffe der objektorientierten Modellierung, beispielsweise der Begriffe Attribut und Methode. Dies ermöglicht es dem Nutzer den Aufbau eines (gut strukturierten) Textverarbeitungsprogrammes zu verstehen. Der Nutzer fragt nun nicht mehr "Wie bekomme ich den Text rot?", sondern lernt, dass rot-sein eine Eigenschaft (ein Attribut) des eines einzelnen Zeichens ist. Diese Art zu denken ermöglicht es ihm, nicht nur funktional zu denken und als "Rechner" auf den Wunsch etwas "rot" zu drucken mit dem trainierten Algorithmus (Format => Zeichen => fett => Farbe => rot => OK) reflexartig zu reagieren, sondern lernt die Struktur zunächst kognitiv zu analysieren (Einzelne Zeichen können rot sein. Das Problem bezieht sich also auf die Formatierung. Also sollte man Format und Zeichen wählen). Dieser Weg ist - obwohl zeitlich aufwendiger - sinnvoller, da so auch in unbekanntem Umgebungen, die richtigen Assoziationen gewählt werden. Durch Naturalisierung können so die Assoziationen ebenfalls schneller umgesetzt werden. Auch die Betrachtung von Tabellen in ihrer ursprünglichen Form, stellt einen erheblichen Vorteil dar. So können Überlegungen auftauchen, dass sich der Rechner sich in irgendeiner Form auch das Ende der Tabelle merken muss. Dies ermittelt einen Einblick in die prinzipielle Funktionsweise, dass Elemente der Struktur nicht irgendwo implizit mit gespeichert werden, sondern, dass die Struktur der Datenmenge in einem bestimmten Format erhalten werden muss. Ein weiteres tiefergehendes Konzept ist bei StarOffice6 die im Klassendiagramm angegebene Möglichkeit zu nutzen Musterdokumente zu erstellen, die selbst wiederum Musterdokumente sind. Dies könnte (abseits von der Nutzung) auf spätere Ideen der Rekursivität wie auch der Selbstanwendung hinweisen.

3. Aufgabe:

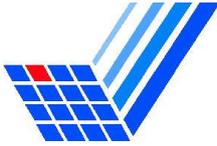
Beide Konzepte legen eine mögliche Grundstruktur für eine Unterrichtseinheit in der Sekundarstufe I an allgemeinbildenden Schulen, in diesem Fall speziell der Real- und Hauptschule, nahe. Das ECDL-Konzept entspricht hierbei weitestgehend den Anforderungen der Wirtschaft und zielt in erster Linie auf Wissen und die Nutzung von konkreten Anwendungen ab, etwas, das vielen Schülern unmittelbaren Nutzen, aber keine längerfristigen Eindrücke bringt. Bei dem ersten Konzept, dem ECDL, handelt es sich um ein Konzept, das seit ca. 10 Jahren umgesetzt wird. Das Ziel besteht in einer Umsetzung auf internationaler (d.h. europäischer) Ebene. Zudem richtet es sich nicht exklusiv an Schulen, sondern auch und vor allem an Menschen, die die Sekundarstufe I bereits absolviert haben. Das Modell ist zudem in vielen Ländern bereits anerkannt und wird staatlich unterstützt. Auf der anderen Seite handelt es sich bei den vorgestellten Konzepten von Siglinde Voß, um Bestrebungen, die aus der Fachdidaktik der Sekundarstufe kommen, also nicht staatlich unterstützt sind. Es handelt sich außerdem, da es sich um ein allgemeinbildendes Konzept handelt, das den Schwerpunkt speziell in der Sekundarstufe ansiedelt, um ein nicht-kommerzielles Konzept. Das erste Konzept hat den Anspruch unabhängig von Softwareherstellern zu sein, während das zweite Konzept exemplarisch die Produkte zweier Softwarehersteller herausgreift: WordXP und StarOffice6. Beide Konzepte zielen darauf ab auch die praktischen Aspekte, die Schüler in der Sekundarstufe I besonders an Realschulen benötigen, direkt zu lehren, ohne andere Inhalte zu wählen und diese über die Argumente einer formalen Bildungstheorie zu begründen. Interessant ist, dass der zweite Ansatz hiermit aufgrund der Idee abhängig von konkreten Produkten wird. Andererseits werden beim zweiten Ansatz allerdings statt der üblichen Klick-Rezepten die Strukturzusammenhänge mittels Objektdiagrammen dargestellt. Ferner befinden sich Klassen- und Sequenzdiagramme in den Unterlagen, die zunächst jedoch nur zur Information des Lehrers gedacht sind. Die Modellierung der abstrakten Zusammenhänge geht deutlich über das hinaus, was der ECDL fordert: es werden Ideen und Konzepte deutlich in den Mittelpunkt gehoben. Der Preis dafür liegt in der Softwareabhängigkeit. Ein weiteres Problem besteht darin, dass das erste Konzept lediglich Zielvorstellungen definiert, aber nicht weiter in die Details der Lehre geht. Es handelt sich also praktisch um Prüfungsunterlagen, die eine Zieldefinition festlegen, die Lehre selbst aber an die entsprechenden Ausbildungsstellen delegieren. Das zweite Konzept stellt momentan im Wesentlichen eine Vorstellung der grundsätzlichen Idee für Lehrer

dar. Es enthält ebenfalls (noch) keine konkreten Unterrichtsbeispiele und Aufgabenbeispiele, die illustrieren könnten, in wie weit das Konzept der Objektorientiertheit im konkreten Unterricht Anwendung findet, d.h. ob es nur als gelegentliches Visualisierungstool genutzt wird oder ob es breite Anwendung findet und vor Allem, ob die Schüler ebenfalls alters- und schulformgemäße objektorientierte Aufgabenstellungen im Rahmen der Unterrichtseinheiten lösen können. In Anbetracht der Tatsache, dass sich die Konzepte derzeit noch in der Entwicklung befinden, fällt es zudem schwer zu beurteilen, ob sie praktisch durchführbar sind oder ob dem entwicklungspsychologische, finanzielle, didaktische oder personelle (auch Lehrer müssen bekanntlich qualifiziert werden) Gründe entgegenstehen. Der ECDL hingegen ist praktisch definitiv durchführbar.

Vorteilhaft ist, dass das Konzept - auch auf der Grundlage weitergehender Qualifizierung des Lehrpersonals - horizontal und vertikal erweiterbar ist. Es könnten Klassen- und Sequenzdiagramme verwendet, analysiert und erstellt werden. Ferner kann dieses Konzept auch auf andere Softwareprodukte (evtl. sogar auf Hardware) ausgedehnt werden. Es ist flexibler und anpassungsfähiger als die rigiden "Klick-Origen", die mittels ECDL gelehrt werden und bietet für die Schüler später im Beruf zudem bei einem Wechsel der Standardsoftware die Möglichkeit, sich die Änderungen durch die Beherrschung des objektorientierten Konzepts anhand geeigneter Diagramme autodidaktisch zu erarbeiten.

4. Aufgabe:

Meiner Ansicht nach stellt das ECDL-Konzept eine gute Möglichkeit dar, grundsätzliche Kenntnisse aus dem Bereich "computer literacy" zu vermitteln, nicht jedoch die wesentlichen Ideen und Konzepte, die ich für ein Schulfach Informatik mit wissenschaftspropädeutischen Inhalten ansehen würde. Das Konzept erscheint zudem, ob seiner Kommerzialität, für den allgemeinbildenden Schulbereich als fragwürdig und didaktisch, methodisch und auch lernpsychologisch wenig untermauert zu sein. Zudem besteht das erwähnte Problem, dass Inhalte zeitlich genauso schnell altern, wie die anwendungsübergreifenden Strukturen der Softwareindustrie. Andererseits ist das Konzept erprobt (seit 1994) und scheint auch durchaus zumindest kurzfristig Erfolge zu zeigen. Das Konzept von Siglinde Voß ist im Vergleich dazu relativ neu. Da das Konzept wie bereits in Aufgabe 3 erörtert erweiterbar ist und zudem mehrere Bereiche aus dem Bereich der Fachwissenschaft Informatik enthält, erscheint es mir angemessener zu sein. Jedoch fehlen derzeit natürlich Übungsaufgaben für die Schüler, die auch die inhaltlichen Anforderungen näher beschreiben könnten und auch erste Auswertungen. (Es handelt sich also momentan faktisch um Material zur Lehrerbildung). Interessant wäre, in wie weit das Konzept faktisch erweiterbar ist und ob die Schüler der entsprechenden Schulformen den objektorientierten Ansätzen zu folgen vermögen. Faktisch fehlen also beim zweiten Ansatz, obwohl er aus sich der Fachwissenschaft Informatik, vielversprechend aussieht, die Verankerung durch empirische Belegung (die ja nach einer E-Mail von Frau Voß allerdings teilweise schon stattfinden) sowie die Erstellung weitergehender didaktischer Materialien.



Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 4

In der Vorlesung wurde im Zusammenhang eines internationalen Vergleichs von Informatikkompetenzen (Stichwort: PISA-Studie) ein Artikel von Hermann Puhlmann¹ genannt, in dem ein theoretischer Rahmen für die Entwicklung von Testaufgaben für die Informatik nach dem PISA-Muster entworfen wird. Die Operationalisierung des Rahmens erfolgt in dem Artikel durch die Angabe von Testaufgaben und ihre Einteilung in eine von drei Kompetenzklassen (Klasse 1: Anwendung, Klasse 2: Gestaltung und Klasse 3: Entscheidung). Es wird in den Testaufgaben – ebenso wie in der PISA-Studie – ausschließlich im Sinne einer Outputorientierung nach prüfbaren Ergebnissen gefragt, also nach den zur Diskussion stehenden Kompetenzen von 15-jährigen Schülerinnen und Schülern und nicht danach, wo diese Kompetenzen erworben wurden.

1. Aufgabe: (8 Punkte)

Entwickeln Sie für jede der drei Kompetenzklassen ein neues Item². Orientieren Sie sich bezüglich des Umfangs und des Schwierigkeitsgrades an den im Artikel aufgeführten Testaufgaben.

Geben Sie darüber hinaus für jedes Item die erwarteten Lösungen an und ordnen Sie es mit kurzer Begründung einer Kompetenzklasse zu.

2. Aufgabe: (2 Punkte)

Geben Sie einen unterrichtlichen Rahmen (inhaltlich und methodisch) an, um die Voraussetzung für eine erfolgreiche Bearbeitung solcher Items zu schaffen.

Abgabe: 12.11.2003, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ Hermann Puhlmann: Informatische Literalität nach dem PISA-Muster, in: Peter Hubwieser (Hrsg.): Informatik und Schule – Informatische Fachkonzepte im Unterricht – INFOS 2003 – 10. GI-Fachtagung 17.–19. September 2003, München. Bonn, S. 59-75

² Insgesamt also drei neue Items.

Anmerkung: Die PISA-Aufgaben aus der 1. Aufgabe sind in einer gesonderten Zusammenstellung als Ergebnis der Lehrveranstaltung „Didaktik der Informatik II“ aufgeführt.

Eine ausgewählte Antwort zur 2. Aufgabe:

Aufgabe 2:

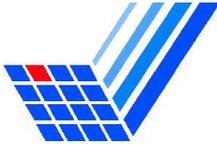
Nach dem Fragenfang in Aufgabe 1 jetzt eine Fangfrage?

Natürlich sind Methoden gerade Inhalte – in der Hoffnung, dass sich daraus Kompetenzen bilden (die Begriffe Inhalt, Thema und Gegenstand sind vielleicht nicht hinreichend definiert). So erwarte ich zum Beispiel nicht, dass alle 15-Jährigen die obige Frage 2 beantworten können, wohl aber, dass sie der Sache nachgehen können und verstehen – an dieser Stelle des Verstehens kann man ruhig die Physik und Biologie in die Pflicht nehmen. Diese Kompetenz lässt sich nicht so einfach durch Items testen, ich halte sie aber für wichtig.

Es stellt sich die Frage, welche Methoden und Kompetenzen in der Sekundarstufe I im Informatikunterricht vermittelt werden sollen. Dies beinhaltet ganz sicher Anwendungskompetenzen und zielt damit nicht darauf, solche Items zu beantworten. Für mich gehören auch ganz klar Analyse-, Strukturierungs- und Modellierungsmethoden der Informatik in die Sekundarstufe I – auf dem Niveau, diese Methoden einsetzen zu können. Zusätzlich kommen investigative Kompetenz aus dem letzten Absatz hinzu und die Kompetenz, Fragen zu stellen – die ja in der zweiten Aufgabe unterstellt wurde.

Anwendungsbeispiel: „Was ist eigentlich Kompetenz?“

Abschließend zum methodischen Vorgehen: Üben, üben, üben. (Mit den Unterrichtsthemen muss dafür gesorgt werden, dass es nicht langweilig wird und auch ein gewisser Grad an Faktenwissen zusammenkommt – aber das gehört ja nicht zur Methodik.)



Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 5

Von Fricke und Völter¹ wird der Ansatz verfolgt, mit Hilfe von Entwurfsmustern die Komplexität des Planungsprozesses für die Vorbereitung, Planung, Durchführung und Reflexion von gestalteten Lernprozessen zu unterstützen. Das entwickelte Schema lässt sich nicht direkt auf die Sekundarstufe I übertragen, da es in erster Linie auf Seminare in der Erwachsenenbildung ausgerichtet ist. Im Folgenden wird der Begriff Pattern mit Muster übersetzt.

1. Aufgabe:

(5 Punkte)

Überprüfen Sie alle 48 Muster des Schemas auf ihre Eignung im Informatikunterricht der Sekundarstufe I. Fassen Sie Ihre Ergebnisse tabellarisch zusammen und geben Sie gegebenenfalls stichwortartig an, warum ein Muster nicht geeignet ist.

2. Aufgabe:

(3 Punkte)

Passen Sie mindestens drei dieser Muster auf den Informatikunterricht in der Sekundarstufe I an. Finden Sie darüber hinaus drei neue Muster und beschreiben Sie sie nach der Vorlage von Fricke und Völter in deutscher Sprache.

3. Aufgabe:

(2 Punkte)

Adaptieren Sie die Übersichtskarte auf die Sekundarstufe I, indem Sie Ihre ggf. umgruppieren, neue Muster hinzufügen und nicht relevante Muster entfernen.

Abgabe: 19.11.2003, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ Fricke und Völter 2000] FRICKE, Astrid ; VÖLTER, Markus: SEMINARS – A Pedagogical Pattern Language about teaching seminars effectively. July 2000. <http://www.voelter.de/data/pub/tp/tp.pdf> oder <http://www.voelter.de/data/pub/tp/html/> – geprüft: 12. November 2003

Anmerkung zu dieser Aufgabe:

Der Ansatz von Fricke und Völter wurde insgesamt recht positiv aufgenommen. Es sind zum Teil recht unterschiedliche Entwürfe erstellt worden, die in einem weiteren Schritt zusammengefasst oder ausdifferenziert werden müssten. Es wurde von Seiten des Übungsleiters vorgeschlagen, in einer weiteren Übung dieses Thema aufzugreifen. Die Studenten entschieden sich im Rahmen für eine Vertiefung in anderen Bereichen, nämlich im Bereich der PISA-Aufgaben und in dem bereits angekündigten paradigmengreifenden Pizza-Beispiel. An dieser Stelle folgen einige Überarbeitungen und Neuentwürfe von Mustern.

Aufgabe 2: (*)

16. NAMEPLATES

Im Informatikunterricht (wie in jedem anderen Unterricht auch) sollten die Lernenden sowohl den Namen des Lehrers sowie die Namen ihrer Mitschüler kennen und als Anrede verwenden. Dies fördert sog. PERSONAL COMMUNICATION und schafft eine angenehme Lernatmosphäre.

Stellen Sie sich Ihren Schülern zu Beginn der ersten Unterrichtsstunde vor (d.h. schreiben Sie ihren Namen an die Tafel o.ä.). Dies erleichtert Lernenden das häufig ohnehin schon "mit Angst belegten" persönliche Zugehen auf den Lehrenden. Durch persönliche Ansprache entsteht auf Lernendenseite, das Gefühl, ernstgenommen zu werden. – Ebenso muss z.B. zu Beginn der Differenzierungsphase im Gymnasium unbedingt bedacht werden, dass sich ggfs. auch die Lernenden noch nicht untereinander kennen, da sie in dieser Konstellation evtl. noch nie Unterricht erhalten haben.

Achtung: Bemühen Sie sich jedoch unbedingt, die Namen der Lernenden auswendig zu lernen. Nach max. 2 Wochen sollten Sie keine Namensschilder mehr benötigen. Eine längere Verwendung (oder auch das tragen sog. "Nametags") scheint eher kontraproduktiv, da beides auf Seiten der Lerner schnell den Eindruck eines desinteressierten Lehrers entstehen lässt, der sich "noch nicht einmal die Mühe gibt, die Namen der Klasse zu lernen".

Sind die Schülerinnen und Schüler der Lerngruppe Ihnen noch unbekannt, empfehlen sich für die ersten Wochen u.U. Namensschilder, die (vorzugsweise mit großer Schrift, damit von Ihrer Position aus lesbar) auf die Schülertische gestellt werden.

44. KIND OF EXAM / CONTENT OF EXAM

Zwar lässt sich im schulischen Kontext die Art der Leistungsüberprüfung nur sehr bedingt beeinflussen, wohl aber der Inhalt derselben. Bauen Sie "Klausuren" so auf, dass diese von den Lernenden nicht von vornherein als "zusätzliches/neues Problem" gesehen werden.

Verwenden Sie in Klausuren oder anderweitigen vorgegeben Prüfungen hauptsächlich Aufgaben, die Lernende nicht mit einer vollkommen neuen Problematik konfrontieren, sondern klar erkennbare Querbezüge zu im Unterricht behandelten Themen und bearbeiteten Aufgaben besitzen. Zwar kann bzw. sollte der in einer Aufgabe anzutreffende Sachverhalt nicht identisch mit unterrichtlich besprochenen Beispielen sein, eine gewisse Affinität scheint jedoch empfehlenswert.

Achtung: Machen Sie nicht den Fehler, ausschließlich die "schwierigen Fälle" bzw. "Ausnahmen" in Klausuraufgaben abzufragen; ein solches Vorgehen resultiert nicht in einem objektiven Bild über den Kenntnis-/Leistungsstand von Schülern (hinsichtlich grundlegender Prinzipien und Methoden), sondern verliert sich im Detail und fragt vielmehr ab, wie gut die Lernenden auswendig lernen können. Leider ist ein solches Vorgehen allerdings noch häufig anzutreffen.

Führen Sie Aufzeichnungen über die im Unterricht verwendeten Aufgaben (z.B. im Rahmen ihrer Unterrichtsplanung). Aus diesem Material können Sie leicht ähnliche Aufgabentypen für eine Leistungsüberprüfung erstellen.

46. SELECTABLE EXAM TIME

Auch wenn der schulische Rahmen der Wahl von Klausurterminen einen engen Rahmen setzt, sollten Sie die Wünsche der Lernenden (in einem gewissen Rahmen) beachten und darauf eingehen.

Stellen Sie sicher, dass eine Leistungsüberprüfung nicht gerade in eine Hochstress- bzw. Entspannungphase der Schülerinnen und Schüler fällt (d.h. Termine direkt vor und nach den Ferien bzw. in der selben Woche wie "viele" andere Prüfungen). Dies kann zum einen durch frühzeitige Planung und Bekanntgabe entsprechender Termine erreicht werden. Die Einhaltung derselben sollte unter Berücksichtigung der Schülerinteressen eher flexibel gehandhabt werden.

Wenn Sie zu Beginn eines Halbjahres einen "Kursplan" erstellen und verteilen, erwähnen Sie auf diesem auch die von Ihnen beabsichtigten Termine für Prüfungen. Dies befreit Sie jedoch nicht von der Notwendigkeit, im unterrichtlichen Rahmen noch mehrfach auf Prüfungen hinzuweisen – von Schülerinnen und Schülern in der Sek.I kann Selbstorganisation/eigene zeitliche Planung nur in einem begrenzten Rahmen erwartet werden.

NEU 1. GEGENSEITIGER RESPEKT

Oft werden (insbesondere kontaktarme) Schülerinnen und Schüler von anderen Lernenden belächelt – ebenso wird ihren Unterrichtsbeiträgen kein Gehör geschenkt. Dies beeinflusst nicht nur besagte Schüler in negativer Art und Weise, sondern schenkt dem Prinzip "NOBODY IS PERFECT" keinerlei Beachtung.

Als Lehrer in der Sek.I sollten Sie hier eingreifend tätig werden. Stellen Sie klar, dass alle Mitglieder der Lerngruppe die gleiche Wertigkeit besitzen. Stellen Sie Denunzierungen als absolut inakzeptabel dar!

Lassen Sie den Beiträgen "schwarzer Schafe" genauso viel Aufmerksamkeit zukommen, wie Beiträgen anderer Mitschüler – selbst, wenn sich innerhalb der Klasse diesbzgl. Widerstand regt. Honorieren Sie die entsprechenden Beiträge!

NEU 2. REALBEZUG

Informatische Prinzipien und Methoden sind selbst für Erwachsene / Sek.II-Schüler abstrakt nur schwer verständlich – selbst, wenn dem ein problemorientiertes Vorgehen zugrunde liegt. Es ist zu erwarten, dass dieses Problemfeld bei Sek.I-Schülern (aufgrund ihrer altersgemäß geringeren Abstraktionsfähigkeit) eine noch größere Dimension besitzt.

Stellen Sie ausreichend häufig Bezüge zwischen Prinzipien und Methoden und realen Problemstellungen her, die idealerweise der Lebenswelt der Lernenden entnommen sein sollten. Dies ermöglicht den Schülern nicht nur ein gründlicheres Verständnis der Anwendbarkeit des "Stoffes", sondern führt ihnen im Umkehrschluss darüberhinaus vor Augen, wie Abstraktion funktioniert.

Es scheint hier angebracht, die Schülerinnen und Schüler selbst über mögliche Anwendungsmöglichkeiten der Gelernten nachdenken zu lassen. Dies kann entweder vollkommen selbständig, etwa im Rahmen von Hausaufgaben, oder aber im Plenum z.B. mittels eines fragend-entwickelnden Unterrichtsgesprächs stattfinden. Idealerweise sollte (was selbstverständlich häufig nicht möglich ist) eines der so erarbeiteten Anwendungsszenarien anschließend von den Schülern zumindest teilweise real umgesetzt werden (z.B. Ampelsteuerung).

NEU 3. COMPUTERCRACKS / COMPUTERNEULINGE

Es ist zu erwarten, dass sich in einem durchschnittlichen Informatikkurs sowohl "Computercracks" (Schüler, die bereits umfangreiche Erfahrungen mit Computern gesammelt haben – evtl. auch bereits eine Programmiersprache beherrschen) als auch "Computerneulinge" (Schüler, für die der Informatikkurs den ersten ernsthaften Kontakt mit dem Medium Computer darstellt) befinden. Leicht kommt es unter diesen Umständen zu einer Monopolisierung des Unterrichts durch die Seite der "Computercracks".

Einer solchen Entwicklung ist dringend entgegenzuwirken, da "Computerneulinge" (die häufig im Hinblick auf das für sie "neue" Medium und die bekannte "Allmacht" ihrer Mitschüler ohnehin verunsichert sein dürften) andernfalls möglicherweise nie ernsthaft zu Wort kommen, um ihre Fragen zu stellen bzw. ihre Lösungsansätze darzubieten. Letzteres ist zwingend notwendig... schließlich richtet sich Informatikunterricht nicht nur an die "Besten" einer Lerngruppe, sondern an den gesamten Kurs.

Beiträge von Computercracks sollten aufgrund ihrer möglicherweise häufigeren Korrektheit nicht zwangsläufig den Beiträgen der übrigen Schüler vorgezogen werden. Sehr gute Schülerinnen und Schüler können sich jedoch in der Korrektur vorliegender Entwürfe/Beiträge hervorragend einbringen (anstatt von vornherein die richtige Lösung zu präsentieren). Hier ist also eine entsprechende Lenkung durch den Lehrer gefragt!.

NEU 97. Think Different – Denke anders

Querdenken kann zu schnellen Ergebnissen und Einsichten führen. Andere Betrachtungsweisen eröffnen neue Zugänge (Muster 20). Und etwas von einer anderen Seite aufzurollen, kann zu neuen und tieferen Einsichten führen. Dies sollte schon in der Planungsphase einer Unterrichtseinheit berücksichtigt werden.

Ermuntere die Schülerinnen und Schüler ihre Sichtweisen und Zugänge darzustellen. Es ist auch möglich, im Unterricht für etwas Gegenbeispiele zu suchen, obwohl bekannt ist, dass es dafür keine gibt. Die Suche und das Ausschließen von Beispielen kann durchaus Einsichten geben.

NEU 98. Take Away Test Fear – Prüfungsangst nehmen

Schülerinnen und Schüler sehen Prüfungen (Klassenarbeiten, Tests, etc.) oft als ein Problem an und haben Angst davor. Dieser Angst muss man begegnen. (Freie Übersetzung der ersten Sätze zu Muster 44. In der Vorlage ist der Prüfungsteil etwas ungeordnet, was in Aufgabe 3 behoben wird.)

Den Schülerinnen und Schülern soll klar sein, dass eine Note nur zur Einschätzung ihrer Leistung dient und nicht zur Einschätzung ihrer Person. Sie sollten die Möglichkeit erhalten, schlechte (Tages-) Leistungen zu kompensieren.

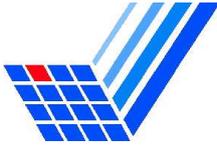
Beispiele dazu sind im Diagramm in Aufgabe 3 aufgeführt.

99. Explicit Expectations – Stelle die Erwartungen klar

Man kann in Prüfungen nur Einfordern, was vorher auch so benannt wurde.

Dies ist ein Gebot der Fairness gegenüber den Schülerinnen und Schülern und verleiht der Lehrer-Schüler-Beziehung Verbindlichkeit.

Vor Prüfungen werden häufig die Prüfungsgegenstände wiederholt, ohne dies explizit zu sagen. Obwohl dies vielen Schülerinnen und Schülern klar sein wird, sollten die Funktionen von Wiederholung und Gegenstände genannt werden.



Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 6

Das „Classroom Rating Instrument“ erlaubt es, im Anschluss einer Unterrichtsbeobachtung eine Beurteilung vorzunehmen. Eine Erweiterung des Instrumentes wurde von F.-W. Schrader und A. Helmke umgesetzt. In [Helmke 2003, S. 283ff]¹ sind die Instruktionen zur Bearbeitung des Ratingbogens abgedruckt.

1. Aufgabe:

(2 Punkte)

Strukturieren Sie die aufgeführten Items zu einem Wissensnetz. Bilden Sie Kategorien, in denen Sie zusammengehörige Items gruppieren können. Verwenden Sie dazu das plattformunabhängige Open-Source-Programm FreeMind². Ihre Abgabe für diese Aufgabe sollte aus der editierbaren Datei bestehen.

2. Aufgabe:

(5 Punkte)

Zeigen Sie exemplarisch an fünf Items Ihrer Wahl die Schulleistung des aktuellen Informatikunterrichts in der Sekundarstufe I (Tagespraktikum) auf. Belegen Sie Ihre Einschätzung mit Zitaten aus den Protokollen der Unterrichtsdurchführung.

3. Aufgabe:

(3 Punkte)

Beurteilen Sie die Einsatzmöglichkeiten und Grenzen des „Classroom Rating Instruments“. Wo und wie kann das Instrument sinnvoll eingesetzt werden? Welche Aspekte, die mit dieser Methode erfasst werden könnten, fehlen in dem oben angegebenen Ratingbogen? Für welche Aspekte würden Sie andere „Erhebungsverfahren“ einsetzen? Begründen Sie ihre Antworten.

Abgabe: 26.11.2003, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ [Helmke 2003] Helmke, Andreas: Unterrichtsqualität erfassen, bewerten, verbessern. Seelze: Kallmeyersche Verlagsbuchhandlung, 2003

² <http://freemind.sourceforge.net/>

Übung "Didaktik der Informatik Sek. I" – Übungsblatt 6 (*)

zu Aufgabe 1.)

Anmerkung: An einigen Stellen war eine eindeutige Kategorisierung schwierig, jedoch habe ich mich bemüht, eine solche zu vollziehen. Einzige Ausnahme bildet das Item: "Lärmpegel in der Klasse", da dieses m.E. in jedem Fall sowohl von der Lerngruppe (Klasse) als auch von der Befähigung des Lehrenden abhängt.

Aufgabe 2.)

Da ich nicht am Tagespraktikum Sek.I teilnehme, habe ich mich für folgendes Vorgehen entschieden, das mir zum einen interessant erscheint, zum anderen der gegebenen Aufgabe sehr nahe kommt:

Bei meiner Lehrtätigkeit (z.Zt. zwei Deutschkurse, jeweils an vier Tagen pro Woche) werde ich regelmäßig durch Vorgesetzte observiert und beurteilt. Dafür wird (ebenfalls) ein standardisierter Ratingbogen verwendet, mit dem folgende Items auf einer Skala von 1 (outstanding) bis 5 (needs a lot of improvement) bewertet werden::

- a) **Instructor is punctual and his teaching is well-organized**
- b) **Organizes and supervises group/student work efficiently**
- c) **Command of German**
- d) **Speaks German only**
- e) **Teaches with much enthusiasm and energy**
- f) **Varies between presentation and student-centered activities**
- g) **Keeps a good pace, doesn't get side-tracked**
- h) **Keeps students' attention at all times, keeps their motivation at a high level (e.g. through the use of well-chosen examples)**

Des Weiteren befindet sich auf dem Fragebogen Platz für "Additional comments", mittels derer entweder nähere Aussagen zu den o.g. Einstufungen getroffen oder zusätzliche Aspekte in die Gesamtbewertung einbezogen werden können. Da mir Einstufungen bzgl. der Punkte dieses Ratingbogens für einige Stunden meines eigenen Unterrichts vorliegen, werde ich diese Daten zur weiteren Bearbeitung dieser Aufgabe verwenden. Dies erspart mir die unobjektive Bewertung meines eigenen Unterrichts – vielmehr werde ich daher die von meinen Vorgesetzten durchgeführten Ratings gemäß der Aufgabenstellung diskutieren (ich nehme an, dass dieses Vorgehen aufgrund meiner speziellen Situation auf Zustimmung trifft).

Im folgenden betrachte ich folgende fünf Items (Anzahl gemäß Aufgabenstellung) des CRI, da sie mit Items des soeben beschriebenen Ratingbogens identisch erscheinen:

| „meine Bildungseinrichtung“ | Äquivalent im CRI |
|---|---|
| Teaches with enthusiasm and energy | 8. Lehrerengagement |
| Organizes and supervises group/student work efficiently, students know who has to do & say what and when | 1. Effektive Regelverwendung |
| Instructor is punctual and his teaching is well-organized | 9. Unterrichtsorganisation |
| Keeps a good pace, doesn't get side-tracked | 10. Lehrstofforientierung |
| Keeps students' attention at all times, keeps their motivation at a high level (e.g. through the use of well-chosen examples) | 15. Bemühung des Lehrers, die Schüler durch interessante Beispiele zu motivieren |

Zunächst die Einstufungen zweier von mir durchgeführten Stunden:

- **Stunde (zu Beginn meiner Tätigkeit):**

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Lehrerengagement | | | | | |
| Effektive Regelverwendung | | | | | |
| Unterrichtsorganisation | | | | | |
| Lehrstofforientierung | | | | | |
| Bemühung des Lehrers, die Schüler durch interessante Beispiele zu motivieren | | | | | |

- **Stunde (vor ca. zwei Wochen):**

| | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Lehrerengagement | | | | | |
| Effektive Regelverwendung | | | | | |
| Unterrichtsorganisation | | | | | |
| Lehrstofforientierung | | | | | |
| Bemühung des Lehrers, die Schüler durch interessante Beispiele zu motivieren | | | | | |

Die Frage, die sich aus diesen beiden Beispielen ergibt, ist geradezu offensichtlich: In welcher Hinsicht waren die Stunden unterschiedlich voneinander, so dass erstere Bewertung mittelmäßig, letztere jedoch ausgezeichnet war? Dies soll im folgenden exemplarisch an einigen den Stunden entnommenen Beispiel analysiert werden. Sofern möglich (d.h. sofern ich mich an solche erinnern konnte), habe ich außerdem Zitate meines Supervisors (Prof. Bruce Spencer) zu den Einzelpunkten angeführt. Verwendet wird zum Vergleich wiederum die Tabellenform.

| 1. Stunde | 2. Stunde |
|--|-------------------------------|
| Lehrerengagement: | |
| <p>Die ausgezeichnete Bewertung resultiert m.E. in beiden Fällen aus meiner großen Motivation für lehrende Tätigkeiten. Ich habe permanent den Wunsch, "mein Wissen an den Mann zu bringen" und ich freue mich enorm darüber, wenn ich andere Menschen von einem Thema begeistern kann. Nach beiden Unterrichtsbesuchen wurde mir mitgeteilt, dass diese meinerseitige "Begeisterung" sich in einem hohen Maß auf die Lernenden überträgt.</p> <p>Zitat eines Vorgesetzten: "They, including myself, could literally feel your enormous interest in the topic – you put so much energy & effort into your teaching that your students can hardly miss its impact."</p> | <p><i>siehe 1. Stunde</i></p> |
| Effektive Regelverwendung | |

| | |
|---|--|
| <p>Es handelt sich hier unschwer erkennbar um die schlechteste Bewertung im Rahmen der Unterrichtsbesuche. Das niedrige Rating basiert auf einiger Verwirrung, die bei der Organisation von Gruppen- und Stillarbeit im Unterricht entstanden ist. Zum einen habe ich die Bearbeitung der gestellten Aufgabe nicht durch "Umhergehen" im Klassenraum überprüft, was dazu führte, dass viele Studenten sich über "Gott und die Welt" unterhalten haben, statt die gestellten Aufgaben zu beantworten. Zum anderen habe ich es versäumt, nach dem Ende der Aktivitäten die Ergebnisse zu überprüfen (z.B. durch Diskussion derselben im Plenum). Aus letzterer Tatsache ergab sich – dies wurde mir anschließend mitgeteilt – ein Gefühl der Hilfslosigkeit auf Studierendenseite, da nicht geklärt wurde, ob die Aufgaben korrekt bearbeitet wurden.</p> | <p>Hier habe ich – als Reaktion auf die für die 1. Stunde erhaltene Kritik – zwei wesentliche Punkte beachtet:</p> <ol style="list-style-type: none"> 1. Aufgabenstellungen waren klar & deutlich und die tatsächliche Bearbeitung wurde durch mich permanent überprüft. 2. Im Anschluss an Gruppen- bzw. Stillarbeitsphasen wurden die Ergebnisse diskutiert und eine korrekte Lösung gemeinsam erarbeitet. |
| <p>Unterrichtsorganisation</p> | |
| <p>Für diese Stunde hatte ich zwar einen detailliert ausgearbeiteten Ablaufplan erstellt – jedoch fehlte eine kurze Auflistung der wesentlichen "Wegpunkte" des Unterrichts für meine Person. Im eigentlichen Unterrichtsgeschehen stellte sich der detaillierte Plan schließlich als wenig hilfreich heraus, da er für einen schnellen Überblick (zu Beantwortung der Frage: "Was wollte ich denn als nächstes machen?") bei weitem zu umfangreich war.</p> | <p>Für diese Stunde hatte ich sowohl einen detaillierten & umfangreichen Plan erstellt (der letztlich dem Beobachter abgegeben wurde) als auch eine Art Stundenskizze (als Referenz für mich). Mit dieser Kombination "floss" der Unterricht besser und wirkte anscheinend allgemein organisierter und durchgeplanter.</p> <p>Zitat meines Vorgesetzten: "Always – even if just mentally – prepare a brief sketch of the lesson you are going to teach."</p> |
| <p>Lehrstofforientierung</p> | |
| <p>Zusammenfassen kann zu diesem Aspekt gesagt werden, dass sowohl Tempo als auch (die damit verbundene!) Lehrstofforientierung nicht optimal waren. Die Studenten schienen während des Unterrichts leicht gelangweilt, da ich – nach Aussage meines Supervisors – zu sehr um das Einbringen von "Anekdoten" bemüht war. Zwar wurden persönliche Erzählungen nicht generell als schlecht klassifiziert, jedoch war es mir wohl nicht gelungen, an dieser Stelle das richtige Maß zu finden.</p> | <p>Die 2. Stunde war wesentlich stringenter am zu behandelnden Lehrstoff ausgerichtet. Zwar war ich noch immer darum bemüht, die Inhalte durch persönliche Erzählungen kreativ "auszuschmücken" bzw. zu ergänzen – allerdings standen fachliche Informationen zu jeder Zeit im Vordergrund meines Handelns.</p> |
| <p>Bemühung des Lehrers, die Schüler durch interessante Beispiele zu motivieren</p> | |
| <p>Diese nur mittelmäßige Bewertung ergibt sich gewissermaßen als Korollar aus der nur schwachen "Lehrstofforientierung". Zwar war ich oft um interessante Beispiel bzw. Erzählungen bemüht, diese passten jedoch oft nicht 100% zum Unterrichtsinhalt.</p> <p>Zitat meines Vorgesetzten: "This may be interesting if you do it once per session – but if you only use examples that have no relation whatsoever with the topics you're covering, your audience gets bored pretty fast."</p> | <p>Wie im Punkt "Lehrstofforientierung" erwähnt, standen stets fachliche Informationen im Vordergrund. In diesem Rahmen gelang es mir, auch die verwendeten Beispiele (real-life examples für Anwendung der deutschen Sprache) entsprechend zielgerichtet auszuwählen. Auf Studierendenseite schien dies die Motivation auf hohem Level zu halten.</p> |

Ergebnisse aus diesen Stunden sind in dieser allgemeinen Form sicherlich auf den Informatikunterricht übertragbar!

Aufgabe 3.)

Zur übersichtlicheren/einfacheren Bearbeitung der Aufgabe verwende ich die Kategorisierung der Items aus Aufgabe 1. Erwähnt sei, dass an hier lediglich der erste "Verzweigungsgrad" der erstellten Mind map referenziert wird. Dies scheint ausreichend detailliert für eine kritische Beurteilung des CRI. Zur besseren Übersicht hier die vollständige Zuordnung in Tabellenform:

| | |
|--|---|
| Befähigung des Lehrenden | 1. Effektive Regelverwendung 2. Vermeidung zeitlicher Fehleinschätzung 3. Vermeidung von Adressierungsfehlern 4. Vermeidung von Überreaktionen 6. Verwenden von Stützmaßnahmen 7. Verständlichkeit von Lehreräußerungen 8. Lehrereengagement 9. Unterrichtsorganisation 11. Verteilung der Antwortgelegenheiten 15. Bemühung des Lehrers, die Schüler durch interessante Beispiele zu motivieren 17. Negative oder herabsetzende Lehrerreaktion bei Schülerantworten 18. Zeit lassen zum Überlegen 19. Sprechweise des Lehrers 20. Ausdrucksweise des Lehrers 21. Wärme, Herzlichkeit 22. Humor 23. Lärmpegel in der Klasse (2fache Zuordnung) |
| Eigenschaften d. Lerngruppe | 12. Beschäftigung nach einer Aufgabenstellung 16. Unterrichtsengagement der Schüler 23. Lärmpegel in der Klasse (2fache Zuordnung) |
| Methodische Qualitätsaspekte des Unterrichts | 5. Aufnehmen von Schülerbeiträgen 10. Lehrstofforientierung 13. Hausaufgabenkontrolle 14. Aufrufen der Schüler nach einem festen Schema |

--Wo und wie kann das Instrument sinnvoll eingesetzt werden?

Grundsätzlich ist der Einsatz des Classroom Rating Instruments (CRI) für die Beurteilung von Unterricht in allen Fächern denkbar - sowohl Sek.I als auch in Sek.II. Obwohl im Artikel selbst lediglich naturwissenschaftliche Fächer an einigen Stellen Erwähnung finden, ist das Konzept m.E. ohne weiteres auf geistes- und gesellschaftswissenschaftliche Fächer etc. übertragbar. Lediglich für künstlerische und Sport scheinen Modifikationen notwendig (hier ist der Unterricht, mehr noch als bei anderen Schulfächern, auf einem individuellen Lernerfolg jedes einzelnen Lerner auszurichten, weshalb das didaktische Vorgehen möglicherweise nicht nach dem gegebenen Schema beurteilt werden kann... dieser Punkt bedarf allerdings weiterer Untersuchung).

Es sei jedoch auf eine allgemeine andersartige Problematik im Zusammenhang mit der Verwendung des CRI hingewiesen: Jegliche Beurteilung von Unterrichtsqualität MUSS sich zwangsweise über einen längeren Zeitraum als der vom CRI-Fragebogen anvisierten einzelnen Stunde erstrecken – auf diesem Sachverhalt wird jedoch an keiner Stelle des zugrundeliegenden Artikels hingewiesen. Die isolierte Beurteilung einer einzelnen Stunde erlaubt keinerlei Aussage über die generelle Qualität des Unterrichts und externe Faktoren – die durchaus immensen Einfluss auf eine einzelne Stunde nehmen können, wie z.B. hohe Temperaturen im Klassenraum –, die mit keinerlei standardisiertem Ratingverfahren zu be-

werten sind. Durch die erwähnte Beobachtung eines längeren Zeitraums kann der Einfluss derartiger Faktoren jedoch erkannt und die Unterrichtsqualität von diesen (nahezu) unabhängig analysiert werden. Für die in diesem Rahmen "observierten" Einzelstunden jedoch scheint das CRI geeignet, wenn auch nur komplementär mit anderen Ratingmethoden, auf die in der folgenden Aufgabenteile eingegangen werden soll.

--Welche Aspekte, die mit dieser Methode erfasst werden könnten, fehlen in dem oben angegebenen Ratingbogen?

Folgende Aspekte sollten m.E. ergänzt werden:

1. weiterführende Beurteilung unterrichtlicher Methodik:

Erstaunlicherweise findet sich im CRI keinerlei Frage bzgl. Sozialformen des Unterrichts. Demnach könnte eine Stunde, die im wesentlichen aus Frontalunterricht und Einzelplatzarbeit besteht, durch eine hohe Wertung erreichen. Dass dem jedoch nicht so sein sollte, ist nach heutigen/modernen didaktischen Gesichtspunkten selbstverständlich. – Zur Erfassung des Aspektes genügt eine recht simple Frage der Art: "Wechselt die Unterrichtsform zwischen Präsentation und schülerzentriertem Arbeiten (z.B. Gruppenarbeit, Partnerarbeit oder Einzelarbeit)?"

2. weiterführende Beurteilung des Schülerverhaltens:

Leider lässt das CRI die Motivationslage der Lernenden weitestgehend außer Betracht. Allerdings ist es im Rahmen verschiedener Klassendynamiken (bei verschiedenen Lerngruppen) selbsterklärend, wie sehr sich eine solche auf die Qualität von Unterricht auswirken kann. Mit per se desinteressierten Schülerinnen und Schülern (solche existieren meiner Meinung nach!) ist schwerlich ein Unterricht von hoher Qualität vorstellbar, während hochmotivierte Lerngruppen das unterrichtliche Geschehen häufig geradezu selbstständig bzw. mit wenig Hilfestellung aktiv gestalten. Vorgeslagene Frage: "Wie motiviert waren die Lernenden, am unterrichtlichen Geschehen aktiv teilzunehmen?"

3. Verwendung der Fachsprache:

Ähnlich den im Rahmen der Fremdsprachendidaktik häufig referenzierten Erkenntnissen aus der SLA-Forschung (second language acquisition) ist auch bei der Herausbildung einer Fachsprache darauf zu achten, dass diese kommunikativ im unterrichtlichen Geschehen Verwendung findet. Werden Lernende hingegen lediglich mit abstrakten Begriffsdefinitionen abgespeist, die in der weitergehenden Kommunikation keine Verwendung mehr finden (z.B. weil der Lehrende der Ansicht ist, er steigere durch Verwendung von Alltagssprache das allgemeine Verständnis), ist ein andauerndes, langfristiges Erinnern an die beschriebenen Fachbegriffe nur schwer vorstellbar. Fragen für das CRI: "Wie sehr bemüht sich der Lehrende um die Verwendung von Fachsprache?" & "Legt der Lehrende darauf Wert, dass auch Lernende Fachsprache verwenden und achtet er auf Korrektheit derselben?".

--Für welche Aspekte würden Sie andere „Erhebungsverfahren“ einsetzen?

Im Sinne einer möglichst umfassenden Antwort auf die gegebene Fragestellung, möchte ich an dieser Stelle keine expliziten Aspekte nennen. Es sei zunächst auf die eingangs beschriebene Kategorisierung der Items hingewiesen. Vor diesem Hintergrund erscheint im wesentlichen die Frage nach weiteren Kategorien für Beurteilungskriterien, die mittels Ratingverfahren nicht oder nur eingeschränkt zu ermitteln sind, sinnvoll.

Es existieren:

- Befähigung des Lehrenden
- Eigenschaften d. Lerngruppe

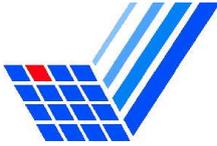
- Methodische Qualitätsaspekte des Unterrichts

Vorstellbar wären z.B.:

- Externe Faktoren (Wetter, Feueralarm, Lautstärke außerhalb des Klassenraums etc)
- Korrektheit (d.h. wie oft/effizient werden Fehler der Lernenden korrigiert, wie "sicher" ist der Lehrende usw.)
- Eindrücke von Lernenden (z.B. Mitschrift von Zitaten)
- Physikalische Realität des Klassenraums/des Kurses (z.B. Sitzordnung)
- Verbesserungsvorschläge(!)
- etc. – dies nur als kleines Ausschnitt; hier sind noch viele Punkte denkbar

Insgesamt erscheint die Beurteilung von Aspekten zu den zuletzt genannten Kategorien nicht im engen Rahmen eines Ratingbogens durchführbar. Vielmehr bietet sich hier ein erweitertes Ratingverfahren an. Der Ratingbogen sollte notwendigerweise um Raum für Freitextkommentare erweitert werden. Zwar ist auf diese Art und Weise die Vergleichbarkeit einzelner Bewertungen nicht mehr im gleichen Maße wie zuvor gegeben, jedoch sind in einem solchen Rahmen nahezu ALLE übrigen Aspekte des Unterrichts zu erfassen. Wo jedoch ist die Grenze zwischen Ratingfähigen / Freitext – Aspekten zu ziehen?

An dieser Stelle ist es notwendig, die Vielfältigkeit der zu bewertenden Aspekte näher zu betrachten. Allgemein sind sicherlich solche Punkte eher durch Ratingverfahren zu beurteilen, die im wesentlichen durch feste Kriterien bestimmt sind (wie z.B. die verwendete Methodik). Für o.g. weitere Kategorien ist die Vielfältigkeit/Variation der zu bewertenden Aspekte jedoch derart groß, dass ein simpler Ratingbogen hier unmöglich ein exaktes Bild des Unterrichts liefern kann – eher würde das Pressen der beschriebenen Aspekte in ein (zu enges) Ratingsverfahren den nachträglichen Eindruck verfälschen. Als Beispiel für diese eher theoretischen Aussagen seien hier "Externe Faktoren" (s.o.) genannt. Es erscheint wenig sinnvoll, diese Kategorie in weitere ratingfähige Unterpunkte einzuteilen, wie etwa: "Wie war die Temperatur im Klassenraum?", "Befand sich vor der Schule eine lärmverursachende Baustelle?", o.ä. – es existieren zu viele derartige (Unter-)punkte. – Im Umkehrschluss betonen diese Erkenntnisse wieder um die Sinnhaftigkeit des beschriebenen Freitextkommentars, mittels dessen deartige Faktoren flexibel und umfassend, aber dennoch knapp zu beschreiben sind (hier etwa: "Die Temperatur im Klassenraum war deutlich zu hoch" bzw. "Vor der Schule befand sich eine Baustelle mit ohrenbetäubendem Lärm").



Übung zu den Vorlesungen „Didaktik der Informatik II und III“

Blatt 7

Im Zusammenhang mit dieser Übung sollen Sie sich mit der Puzzle-Methode¹ vertraut machen. Dabei handelt es sich um eine Kombination von Gruppenarbeit und autonomen Lernen. Die Inhalte werden in mehrere, voneinander unabhängige Themenbereiche aufgeteilt, die jeweils von einer Expertengruppe bearbeitet werden. Die Expertengruppen werden nach dieser Phase aufgelöst und es werden Unterrichtsrunden gebildet, in denen jeweils genau ein Experte eines Themenbereichs beisitzt.

Sie werden in der Übung am 27.11. einer von vier Expertengruppe zugeteilt (1. Gruppe: objektorientiert, 2. Gruppe: funktional, 3. Gruppe: Logik (wissensbasiert), 4. Gruppe: Datenfluss).

In der Übung am 4.12. werden die einzelnen Elemente der Unterrichtsreihe mit dieser Methode praktisch durchgeführt und im Anschluss reflektiert.

1. Aufgabe: (4 Punkte)

Im kollaborativen Arbeitsbereich zur Veranstaltung finden Sie die Materialien für diese Übung². Lesen Sie die Dokumente `A1-Einstieg.pdf`, `A2-imperative-Loesung.pdf` und bearbeiten Sie die für Ihre Gruppe spezifischen Aufgaben. Überprüfen Sie anschließend Ihre Lösung mit der für das Selbststudium vorbereiteten Lösung (wird auf Anfrage per E-Mail verschickt).

2. Aufgabe: (6 Punkte)

Bereiten Sie sich **schriftlich** auf die Unterrichtsrunde vor.

Beachten Sie die in dem Dokument `M-Mini-Didaktik.pdf` angegebenen Tipps.

Abgabe: 3.12.2003, 12⁰⁰ Uhr

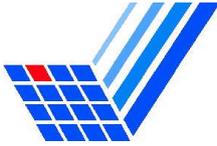
Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ siehe: <http://www.educeth.ch/didaktik/puzzle/>

² Das dieser Übung zu Grunde liegende Beispiel findet sich in den Materialien der ETH Zürich zum Informatikunterricht (vgl. [Brennwalder und Stamm 1994] BRENNWALDER, Daniel ; STAMM, Christoph ; HARTMANN, Werner (Hrsg.): Gruppenunterricht zum Thema: Paradigmen von Programmiersprachen. Zürich : ETH, September 1994. – ETH – Eidgenössische Technische Hochschule Zürich – Institut für Verhaltenswissenschaft / Departement für Informatik PDF-Dokument vom 6. November 1997 <http://educeth.ethz.ch/informatik/puzzles/paradigmen/> – geprüft: 14. Juli 2003)

Anmerkung zu dieser Übung:

Die Übung wurde gemeinsam mit den Teilnehmerinnen und Teilnehmer der Veranstaltung DDI 3 durchgeführt. Dabei wurden Unzulänglichkeiten in dem Ausgangsmaterial aufgedeckt, die aufgearbeitet werden mussten. In den Übungen 10 und 11 wurden die Materialien überarbeitet (siehe unten). Die Ergebnisse werden in einem separaten Dokument zusammengestellt.



Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 8

In der Vorlesung wurde das Buch „How to Think Like a Computer Scientist – Learning with Python“ von Gregor Lingl¹ vorgestellt. Durch die ausgeprägte Ausrichtung an der Fachsystematik eignet es sich in erster Linie für „die Hand des Lehrenden“ und weniger für eine direkte Übertragung in den Informatikunterricht in der Schule. Ihre Aufgabe ist es im Rahmen dieser Übung, angemessene Beispiele für den Informatikunterricht in der Sekundarstufe I zu entwickeln.

1. Aufgabe:

(3 Punkte)

Welche Fachkonzepte, die in dem oben genannten Buch thematisiert werden, eignen sich für die Sekundarstufe I? Fertigen Sie eine Concept Map an, in der die fachlichen Abhängigkeiten ersichtlich werden.

2. Aufgabe:

(7 Punkte)

In Kapitel 6 „Iteration“ werden zur Veranschaulichung Logarithmen-Tabellen und Multiplikationstabellen eingeführt, die – wie der Autor bemerkt – heutzutage nicht mehr sonderlich nützlich sein mögen². Da Logarithmen-Tabellen weder in der Hauptschule noch in der Realschule, noch in der Gesamtschule oder im Gymnasium thematisiert werden und sie eher der Mathematik zuzuordnen sind, ist es angebracht, geeignetere Beispiele zu entwickeln.

- Finden Sie andere Beispiele (möglicherweise auch nur ein „durchgehendes Beispiel“), anhand derer Sie die fachlichen Konzepte der Kapselung und der lokalen Variablen thematisieren können und beschreiben Sie sie umgangssprachlich. (2 Punkte)
- Geben Sie Aufgabenstellungen an, die aus Sicht der Lernenden interessant zu lösen sein könnten, um Kapselung und lokale Variablen thematisieren. (1 Punkt)
- Arbeiten Sie Ihre Beispiele zu lauffähigen Python-Programmen aus. (4 Punkte)

Abgabe: 10.12.2003, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ Bezugsquelle: <http://www.greenteapress.com/thinkpython/> (geprüft am 2.12.2003)

Eine deutsche Übersetzung der ersten sechs Kapitel:

<http://ada.rg16.asn-wien.ac.at/~python/how2think/index.htm> (geprüft am 2.12.2003)

² Originalzitat: „Although a “log table” is not as useful as it once was, it still makes a good example of iteration.“

Übungszettel 8 (*)

Aufgabe 1

Anmerkung: Beiliegende Concept-Map ist sehr(!) umfangreich. Jedoch bin ich der Meinung, dass prinzipiell alle aufgeführten Aspekte in der Sekundarstufe I - zumindest in Ansätzen - vermittelbar sind. Sicherlich ist es jedoch NICHT möglich, alle Punkte in einem einzigen Sek.I-Kurs zu behandeln! Die Kapitel 17-19 sowie den Anhang habe ich in meiner Aufstellung nicht berücksichtigt.

Die darin vermittelten Konzepte übersteigen meiner Meinung nach die maximal möglichen Anforderungen im Rahmen des Sek.I-Unterrichts.

Aufgabe 2

ACHTUNG: Ich bin bei meiner Lösung davon ausgegangen, dass Python Parameter an Funktionen mittels Call by value übergibt - so steht es jedenfalls in diversen Tutorials! Sollte diese Information falsch sein, so sind es auch einige meiner Lösungen... leider gibt das Buch darüber keinen wirklichen Aufschluss.

Aufgabenteil a)

Generell bin ich hier für eine vollkommen andere (einfachere) Vorgehensweise als die im Buch anzutreffende. Zwar fügen sich die Beispiele, in diesem Fall Logarithmstabellen, gut in den Kontext des restlichen Kapitels ein (es wird damit "ein roter Faden" fortgesetzt) soll jedoch ein exklusiver Schwerpunkt (z.B. im Rahmen einer Unterrichtssequenz) auf die Themen "Kapselung" und "lokale Variablen" gelegt werden, bieten sich einfacher strukturierte Beispiele m.E. ohnehin an.

Für beide Konzepte wird im folgenden ein durchgehendes Beispiel verwendet (welches jedoch aufgrund einer kleinen sprachlichen „Ungenauigkeit“ noch durch ein anderes komplementiert wird). Der ebenfalls gemäß Aufgabenstellung zu schreibende Pythoncode ist bereits hier zu finden. Im übrigen habe ich mich entschlossen, die Aufgabe so zu bearbeiten, als würde ich einen Abschnitt des Buches mit meinen Beispielen (und mit Schwerpunkt auf die Sek.I neu schreiben).

- Als Beispiel zu Kapselung würde ich eine einfachere Funktion verwenden, die lediglich den Sinn von Kapselung demonstriert. Erinnerung sei an die im ersten Kapitel des Buches besprochene Funktion `newLine()`. Zunächst kann ein einfacher Algorithmus betrachtet werden, der eine beliebige Anzahl n an Leerzeilen auf dem Bildschirm ausgibt:

```
def newLine():
    print "\n"

prompt = "Bitte geben Sie die Zahl der Leerzeilen ein: "
zahl=input(prompt)
zaehler=zahl;
while zaehler>0:
    newLine()
    zaehler=zaehler-1

print "\nFertig!" + zahl + "Leerzeilen geschrieben!"
print "Fertig!"
```

- Zur Verdeutlichung des Prinzips der Kapselung, wird dann die while-Schleife in eine Funktion "gesteckt" (Zitat Buch: "Encapsulation is the process of wrapping a piece of code in a function, allowing you to take advantage of all the things functions are good for.", p. 65). Es sollte im Unterricht unbedingt nochmals darauf hingewiesen werden, welche Vorteile Funktionen gegenüber "direkter" Implementierung haben (z.B. bessere Lesbarkeit, weniger Schreibaufwand, usw.). Die Verwendung einer solch einfachen Funktion zur Demonstration des Prinzips der Kapselung hat den entscheidenden Vorteil, dass das Verständnis auf Lernendenseite nicht durch andere Aspekte eines evtl. komplizierteren Codes erschwert wird.

```

def newLine():
    print "\n"

def newLines(zahl):
    while zahl>0:
        newLine();
        zahl=zahl-1

prompt = "Bitte geben Sie die Zahl der Leerzeilen ein: "
zahl=input(prompt)
newLines(zahl)

print "\nFertig!" + zahl + "Leerzeilen geschrieben!"

```

- An obigem (sehr einfachen, aber gerade deshalb aussagekräftigen) Beispiel kann im Anschluss auch das Prinzip der "lokalen Variablen" erläutert werden. Zwar handelt es sich bei `zahl` strenggenommen um einen Parameter, jedoch lässt sich das Prinzip der Lokalität damit ebenfalls beleuchten. Es fällt bei genauer Betrachtung auf, dass im zweiten Beispiel sowohl `newLines()` als auch das Hauptprogramm eine Variable `zahl` verwenden. Es stellt sich somit die Frage, ob dies zu Problemen führt, da `zahl` in `newLines()` heruntergezählt wird, die Hauptprogramm-Variable "zahl" jedoch nach Beendigung der Funktion noch für eine Ausgabe benutzt wird. Es ist für uns an dieser Stelle selbstverständlich offensichtlich, dass es aufgrund von Lokalität nicht um die selbe Variabel handelt. In Beispiel Nr.1 (s.o.) musste der Wert aus `zahl` zu diesem Zweck übrigens explizit in eine zweite Variable übernommen werden.
- Damit jedoch auch noch - dem Buch entsprechend - ein „echtes“ Beispiel zu „lokalen Variablen“ zu sehen ist, hier ein kleines Pythonprogramm, welches ich ebenfalls aufgrund seiner Einfachheit zur Demonstration des Phänomens für gut geeignet halte. Es stellt aus zwei Namenslisten (Vornamen & Nachnamen) per Zufallsauswahl einen vollständigen Namen (z.B. für's Kind ;-)) zusammen. Die Bedeutung von Lokalität ergibt sich in dem Moment, wo `waehle_name()` die Funktion `waehle_nachname()` aufruft, da beide die Variable `index` benutzen.

```

def waehle_nachname():
    import random
    nachnamen=["Meier", "Papoulias", "Krüger", "Schmidt"]
    index=random.randrange(0, len(nachnamen))
    return(nachnamen[index])

def waehle_name():
    import random
    vornamen=["Heribert", "Karl", "Marius", "Athanasios"]
    index=random.randrange(0, len(vornamen))
    return(vornamen[index] + " " + waehle_nachname())

print "Der ausgesuchte Name für ihr Kind ist: " + waehle_name()

```

- Sicherlich sind kompliziertere Beispiele DENKBAR, jedoch halte ich die obigen kleinen Beispielprogramme zur Einführung in die Thematik für ausreichend.

Aufgabenteil b)

Mögliche Aufgabe 1:

Markus ist Informatikschüler an einer Realschule in Radevormwald. Zur Zeit ist die Programmiersprache "Python" das Unterrichtsthema. Markus versucht schon seit einiger Zeit, eine Funktion zu schreiben, die die Inhalte zweier Variablen miteinander vertauscht, aber irgendwie "will es nicht so richtig klappen".

Beide PRINT-Befehle (s.u.) liefern das selbe Ergebnis. Wo liegt das Problem bei seinem Programm? Beschreibe den Fehler so ausführlich wie möglich!

```
def tausche(a, b):
    retter=a;
    a=b;
    b=retter;

a=input("Bitte gib einen Wert für a ein: ")
b=input("Bitte gib einen Wert für b ein: ")

print "\n\na ist gleich "+ a + "und b ist gleich " + b + "\n"
tausche(a,b)
print "a ist gleich "+ a + "und b ist gleich " + b
```

Mögliche Aufgabe 2: Betrachte folgendes Programm in Python:

```
a=input("Bitte gib eine Basis ein:")
b=input("Bitte gib einen Exponenten ein:")
c=input("Bitte gib einen zweiten Exponenten ein:")

zwischen_wert=1
retter=b

while retter>1:
    zwischen_wert=zwischen_wert*a
    retter=retter-1

wert=zwischenwert
retter=c

while retter>1:
    wert=wert*zwischen_wert
    retter=retter-1
```

- a) Was macht das Programm?
- b) Wie kann das Programm durch Kapselung verbessert werden? Lässt sich vielleicht ein Teil in einer Funktion zusammenfassen? Dürfen die beiden Funktionen gleichlautende Variablen besitzen, oder würden sich diese "stören"?

Mögliche Aufgabe 3 (Zusatzaufgabe): Der Informatiklehrer Nitram Ztrenier zermartert sich das Hirn. Er hat folgendes herausgefunden: Das folgende einfach Python-Programm...

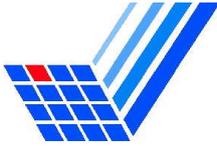
```
def zaehle(n):
    zaehler=1
    while zaehler <= n:
        print zaehler + " ",
        zaehler=zaehler+1
    zaehle(100)
```

...lässt sich auch so schreiben:

```
def r_zaehle(n):
    if n>1:
        r_zaehle(n-1)
    print n + " ",
    r_zaehle(100);
```

Allerdings hat Herr Rekur leider gar keine Ahnung, warum das so ist. Vielleicht kannst du ihm ja helfen.

- a) Zeichne ein Stack-Diagramm über die Aufrufe von #r_zaehle#! Welchen Wert hat n jeweils?
- b) Hast du eine Idee, was genau das zweite Programm macht und warum es funktioniert?



Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 9

Dieses Übungsblatt schließt an die vierte Übung an, in der Entwürfe für PISA-Aufgaben entwickelt wurden. Eine Zusammenstellung der von Ihnen entwickelten Aufgabenentwürfe sowie eine Auswahl von PISA-Aufgaben für mathematische Kompetenzen finden sie im kollaborativen Arbeitsbereich.

1. Aufgabe: (5 Punkte)

Erweitern Sie die Entwürfe nach dem Muster der mathematischen PISA-Aufgaben. Eine genaue Zuordnung der zu überarbeitenden Aufgaben findet in der Übung am 11.12. statt¹. Die Aufgaben sollten nach der Überarbeitung als Überschrift einen Beispielkontext enthalten, die Aufgabenstellung und die Arbeitsanweisung sollte verständlich formuliert sein und eine Bewertung nach folgendem Muster enthalten:

Bewertung

Full Credit

Code 2: (Beschreibung der vollständigen Lösung, ggf. mit Beispielen)

Partial Credit

Code 1: (Beschreibung von teilweise korrekten Lösungen mit Beispielen)

No Credit

Code 0: Andere Antworten (ggf. inklusive Antworten ohne Begründung)

Code 9: Missing.

Falls es keine teilweise korrekten Lösungen gibt, entfällt der Punkt „Partial Credit“ und der Punkt „Full Credit“ erhält den „Code 1“.

2. Aufgabe: (5 Punkte)

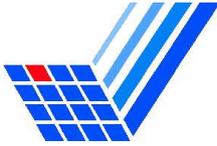
Entwickeln Sie nach dem gleichen Schema eine weitere Aufgabe mit drei Fragen aus einem neuen Beispielkontext und entwickeln Sie dazu sowohl eine Musterlösung, wie auch die Bewertung.

Abgabe: 17.12.2003 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ Es handelt sich um 5 bis 6 Aufgaben pro Person.

Anmerkung: Die in dieser Übung entwickelten PISA-Aufgaben werden in einem separaten Dokument in einer überarbeiteten Fassung veröffentlicht.



Übung zur Vorlesung „Didaktik der Informatik II und III“

Blatt 10 und 11

In der Besprechung zur Übung zum Übungsblatt 7 hat sich herausgestellt, dass das „Pizza-Beispiel“ zum Thema „Paradigmen von Programmiersprachen“ mit der Puzzle-Methode von [Brennwalder und Stamm 1997]¹ für den Einsatz in der Sekundarstufe II überarbeitet werden sollte. Diese Überarbeitung sollte weiterhin auf eine (einzige) Programmiersprache hin ausgerichtet werden – hierfür ist die Programmiersprache Python besonders geeignet² –, um Unterschiede und Gemeinsamkeiten der Paradigmen deutlich herausstellen zu können.

Mit diesem Übungsblatt erhalten Sie die Aufgabe, die Materialien arbeitsteilig (in den im 7. Übungsblatt gebildeten Teilgruppen) für Ihr Paradigma zu überarbeiten.

1. Teil (bis zum 7.1.2004):

Aufgabe 1:

(3 Punkte)

Recherchieren Sie nach Lehrbüchern und weiteren Quellen, die für ihr Paradigma nützlich sind. Wählen Sie ein oder zwei Lehrbücher aus und nennen Sie auch die weiteren Quellen mit einer kurzen Beschreibung.

Aufgabe 2:

(7 Punkte)

Programmieren Sie in Python eine Implementation des Pizza-Beispiels für Ihr jeweiliges Paradigma. (Zusatzaufgabe für die OO-Gruppe: ... und für das imperative Paradigma)

2. Teil bis zum 14.1.2004:

(10 Punkte)

Überarbeiten Sie für Ihr jeweiliges Paradigma die Arbeitsblätter für das Selbststudium und die Lösungsblätter für das Selbststudium.

Abgabe 1. Teil: 7.1.2004, 12⁰⁰ Uhr

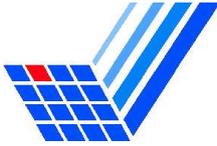
Abgabe 2. Teil: 14.1.2004, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ [Brennwalder und Stamm 1997] BRENNWALDER, Daniel ; STAMM, Christoph ; HARTMANN, Werner (Hrsg.): Gruppenunterricht zum Thema: Paradigmen von Programmiersprachen. Zürich : ETH, November 1997. – ETH – Eidgenössische Technische Hochschule Zürich – Institut für Verhaltenswissenschaft / Departement für Informatik PDF-Dokument vom 6. November 1997
URL: <http://educeth.ethz.ch/informatik/puzzles/paradigmen/> – geprüft: 17. Dezember 2003)

² Vergl.[Linkweiler 2002] LINKWEILER, Ingo: Eignet sich die Skriptsprache Python für schnelle Entwicklungen im Softwareentwicklungsprozess? Diplomarbeit, FB Informatik, Universität Dortmund, 16. November 2002 URL: <http://www.icc-computer.de/ingo/diplom/>

Anmerkung: Die Zusammenstellungen zu dem Paradigmen-Pizza-Beispiel werden in einem separaten Dokument aufgeführt.



Übung zur Vorlesung „Didaktik der Informatik III“

Blatt 12

Dieses abschließende Übungsblatt der Lehrveranstaltung „Didaktik der Informatik III (für Sek I)“ soll Ihnen die Gelegenheit bieten, Stellung zu beziehen zu ihrer persönlichen Vorstellung eines „guten Informatikunterrichts“.

Aufgabe 1:

(3 Punkte)

Welche Inhalte halten Sie persönlich für die Sekundarstufe I für wesentlich? Differenzieren Sie hierbei nach unterschiedlichen Altersstufen. Ordnen Sie darüber hinaus die genannten Teilgebiete den Fachgebieten zu, wie Sie sie z. B. aus der (universitären) Fachsystematik her kennen.

Aufgabe 2:

(3 Punkte)

Welche Methoden sollen in Ihrem Informatikunterricht zum Einsatz kommen? Zeigen Sie die lerntheoretischen Grundlagen Ihrer bevorzugten Methoden auf und beschreiben Sie, in welchem Umfang sie welche Methode einsetzen möchten. Stellen Sie dabei ggf. auch Bezüge zu den in Aufgabe 1 genannten Inhalten her.

Aufgabe 3:

(4 Punkte)

Stellen Sie eine Konzeption Ihres Informatikunterrichts dar. Beschreiben Sie das Verhältnis zwischen Lehrern und Schülern. Welchen Stellenwert hat Ihre Halbjahresplanung, Planung der Unterrichtsreihe und Planung der Unterrichtsstunde? In welchem Verhältnis stehen die Inhalte zur Methode? In welcher Form findet Leistungsbewertungen statt? Wodurch unterscheidet sich eine „gute“, eine „befriedigende“ oder „ausreichende“ Leistung der Schüler? Inwieweit lässt sich Ihr „Modell“ auf den Unterricht ihrer (vorgestellten) Fachkollegen übertragen?

Abgabe: 21.1.2004, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

Aufgabe 1:

Verschiedene Inhalte sind für die Informatik der Sekundarstufe I relevant und sollten wenn möglich behandelt werden. Durch den eingeschränkten Zeitrahmen, welcher im schulischen Kontext zur Verfügung steht, werden jedoch sicherlich Abstriche gemacht werden müssen.

Folgende Inhalte sind nach Möglichkeit, und auch der Interessenlage der Lernenden, zu berücksichtigen:

1. Grundlagen der Informatik

- (a) Die historische Entwicklung der Informatik oder der Informationsverarbeitung im allgemeinen sollte angesprochen werden. Dazu gehört auch das Prinzip, dass Daten auf verschiedene Weise dargestellt werden können. So z.B. als Zeichenkette, als Dezimalzahl, in Keilschrift, in römischen Ziffern und letztlich als Binärzahlen.
- (b) Das Grundprinzip nach dem Daten verarbeitet werden (EVA) sollte angesprochen werden.

2. Rechnerstrukturen

- (a) Weiterhin sollte darauf eingegangen werden, wie dieser Verarbeitung im Computer aussieht. Das bedeutet einfaches Rechnen mit Binärzahlen und die Grundlagen der Logik. Ein einfaches Schaltwerk könnte angesprochen werden.
- (b) Die Struktur eines Einzelplatzrechners sollte angesprochen werden. Das bedeutet das Prinzip des von-Neumann-Rechners zu verdeutlichen. Aber auch die verschiedenen Komponenten die nicht unbedingt im stark abstrahierten Modell des von-Neumann-Rechners vorkommen sollten durchgenommen werden. So z.B. Festplatte, Grafikkarte, Co-Prozessor usw. Es kann auch kurz auf alternative Konzepte eingegangen werden (z.B. Analogrechner).

3. Informationssysteme

- (a) Das Schulnetz muss vorgestellt werden. Eine einfache Darstellung seiner Funktionsweise und seines Aufbaus sowie seiner Benutzung sollte gegeben werden. Verschiedene andere Netzwerkvarianten könnten angesprochen werden. Natürlich muss beim Stichwort Netzwerk auch das Internet und seine Konzeption in einfacher Weise angesprochen werden.

4. Softwaretechnik

- (a) Es sollte herausgestellt werden, wie ein Informatiksystem von der Softwareseite aus aufgebaut ist. Dazu gehört die Unterscheidung zwischen Betriebssystem und Anwendungssoftware und deren Aufbau. Der Punkt Anwendungssoftware sollte durch verschiedene Beispielanwendungen untermauert werden. Auch die Entwicklung von Software sollte angesprochen werden. Programmiersprachen und ihre Unterscheidung in maschinennahe und höhere sollte herausgestellt werden.
- (b) Es sollte anhand einer Pseudoprogrammiersprache oder noch besser eine didaktisch geeigneten realen Programmiersprache wie Python herausgearbeitet werden, wie, im objektorientierten Paradigma, ein Programm aufgebaut ist. Das Prinzip der Problemlösung in Form von Algorithmisierung sollte vermittelt werden. Dabei finden auch bereits einfache Standardalgorithmen wie die allseits bekannten Such- und Sortierverfahren ihre Anwendung.
- (c) Es sollten die beiden wichtigsten Programmierparadigmen verdeutlicht werden. Imperatives Programmieren und objektorientiertes Programmieren. Einfache Modellierungsaufgaben sind denkbar. Wichtig ist dabei, mit dem objektorientierten Paradigma zu beginnen, damit sich eine der gestaltete informatische Weltansicht einprägt. Ein kurzer Überblick über die beiden anderen Paradigmen wäre wohl auch wünschenswert.

5. Theoretische der Informatik

- (a) Ein einfacher Einstieg in den Bereich der Komplexität und des Laufzeitverhaltens sollte auch hier schon vermittelt werden. Dabei muss nicht auf korrekte formale Notationen Wert gelegt werden. Der Fokus sollte auf dem Entwickeln eines Gefühls für „naive“ und „schlaue“ Algorithmen liegen.

6. Informatik und Gesellschaft

(a) In diesem Punkt geht es in erster Linie um Datenschutz und Datensicherheit, sowie die Auswirkungen der Informationstechnologie auf unsere Gesellschaft. Das Stichwort „Verantwortung“ ist hier entscheidend.

7. Modellierung und Programmierung

(a) Die Schüler und Schülerinnen sollten in der Lage sein, einfache Lösungen in Pseudocode oder einer didaktisch eingängigen Programmiersprache (z.B. Python) zu erstellen.

(b) Darüber hinaus sollte eine objektorientierte Modellierung eines einfachen Problembereiches möglich sein.

Aufgabe 2:

Im Folgenden sollen einige Methoden genannt werden, welche in einem Informatikunterricht der Sekundarstufe I eingesetzt werden könnten:

1. Als eine der wichtigsten Arbeitsmethoden der Informatik ist die Teamarbeit bzw. Gruppenarbeit zu nennen. Sie sollte auf jeden Fall vermittelt und natürlich auch angewandt werden, sofern dies im jeweiligen Unterrichtskontext möglich ist. Dies sollte aber in einer Vielzahl der Situationen möglich sein.
2. Einige der oben genannten Inhalte könnten durchaus auch von Schülern in Form von Referaten bearbeitet werden. Die Methoden des wissenschaftlichen Arbeitens und der Recherche sollten also auch ein Thema des Unterrichts sein.
3. Exploratives Lernen mit Hilfe von entsprechenden Lehr-/Lernsystemen könnte ebenfalls eine vielversprechende Methode darstellen, sofern dafür geeignete Softwareprodukte zu finden sind.
4. Simulations- und Rollenspiele sind auch in der Informatik möglich. So könnte z.B. die paketvermittlungsbasierte Datenübertragung in Netzwerken mittels eines solchen Simulationsspiels von den Lernenden nachvollzogen werden. Einen entsprechenden Versuch konnte ich in einem meiner Praktika bereits mit Erfolg durchführen.
5. Natürlich werden auch klassische Arbeitsmethoden wie die Partnerarbeit, die Stillarbeit, der Lehrervortrag usw. im Informatikunterricht vorkommen und auch müssen.

Aufgabe 3:

Bei der Konzeption eines befriedigenden Informatikunterrichts in der Sekundarstufe I, aber auch in späteren Jahrgängen, sollte zwei Aspekte im Vordergrund stehen. Zum einen sollten die wichtigsten informatischen Inhalte vermittelt werden, was eine stringente Planung und somit auch deren Befolgung in Form von Halbjahresplänen und Plänen von Unterrichtsreihen erfordert. Zum anderen sollten aber auch die jeweiligen Interessen der Lernenden, die sich nur im unwahrscheinlichen Idealfall mit der Planung des Lehrenden decken, berücksichtigt werden. Daher bedarf es in der Planung einer ausreichend bemessenen Toleranz zur Einbindung solcher Schülerwünsche.

Ein weiterer entscheidender Punkt ist die Bewertung von Schülerleistungen. Am einfachsten gestaltet sich da eine schriftliche Leistungserfassung in Form einer Klausur oder eines Testes. Eine solche Bewertung erscheint aus Gründen der größeren Objektivität und gleichmäßigen quantitativen Verteilung bei den Lernenden legitim und sinnvoll.

Aber auch andere Alternativen der Leistungsbewertung sollte es geben. So z.B. die Bewertung von Referaten (gegebenenfalls auch unter Mitwirkung der anderen Lernenden), von Projektarbeiten und Hausarbeiten. Auch die klassische mündliche Beteiligung darf, trotz der Kritik ob ihrer Subjektivität, nicht vergessen werden.

Übungsblatt 12 (*)

Aufgabe 1.)

Achtung:

Die folgende Auflistung erhebt keinerlei Anspruch auf Vollständigkeit, sie soll vielmehr als grobe Orientierung dienen, welche Themen im Informatikunterricht einzelner Jahrgangsstufen meines Erachtens unabdingbar sind. Die Aufstellung ist lediglich als Grundfundus zu sehen, der Schülerinnen und Schülern der verschiedenen Jahrgangsstufen vermittelt werden sollte. Selbstverständlich muss der konkrete schulische Informatikunterricht zwingend um weitere Inhalte bereichert werden.

[Zuordnung zur Fachsystematik]

Jahrgangsstufe 5/6:

- Aufbau einer Computer Literacy grundlegende Bedienung eines Computers und von einfacher Software (Texteditor, Textverarbeitung), sowie Kenntnis der wichtigsten Hardware [Computer Literacy, praktische Informatik, teilw. Rechnerstrukturen auf sehr niedriger Ebene]
- erster Kontakt mit grundlegenden informatischen Strukturen (an konkreten Beispielen!), wie [theoretische Informatik & Logische Grundlagen]:
 - Klammerstrukturen
 - Logik
- Kommunikation als wesentliches Element von Informatiksystemen [theoretische Informatik - Informationssysteme, Informatik und Gesellschaft]
- Datenschutz, Recht auf informatische Selbstbestimmung [Informatik und Gesellschaft]
- verantwortungsvolle Nutzung des Internet [Informatik und Gesellschaft]
- statische Modellierung in Anfängen (z.B. Aufbau von HTML-Dokumenten) [praktische Informatik]

Jahrgangsstufe 7/8:

- Ausweitung der Computer Literacy auf kompliziertere Software wie Tabellenkalkulation, Datenbank, etc. [computer literacy]
- Ausweitung informatische Strukturen, z.B. auf [theoretische Informatik]:
 - Variablen
 - konditionale Verzweigung
 - Schleifen
- Modellierung von Prozessen [prinzipiell zu allen Bereichen der Informatik gehörig]
- Boolesche Algebra in Anfängen [mathematische & Logische Grundlagen]
- Fragen bzgl. (II)legalität von eigenem Handeln in öffentlichen Netzen [Informatik und Gesellschaft]

Jahrgangsstufe 9/10:

- Paradigmen von Programmiersprachen [praktische Informatik]
 - unterschiedliche Modellierungskonzepte
- Programmiersprachliche Umsetzung der bisher erworbenen informatischen Strukturen [praktische Informatik]
- Erweiterung informatische Strukturen [theoretische Informatik]:
 - verschiedene Speicherarten (z.B. Stacks) und Zugriffsprinzipien (z.B. FIFO)
 - dynamische Datenstrukturen (Baum, Liste, etc.)
 - einfache Effizienzüberlegungen
- Vernetzung / Netzwerke [praktische Informatik]
- Logische Schaltungen (Aufbauend auf zuvor gebildeten Logischen Grundlagen) [praktische Informatik]
- Realisierung von Datensicherheit u. informatischer Selbstbestimmung [Informatik und Gesellschaft]

Es sei darauf hingewiesen, dass die von mir konstituierten notwendigen Elemente des Informatikunterrichts prinzipiell eine Synthese aus im Rahmen der DDI 3-Vorlesung behandelten Ideen zur informatischen Bildung ist (so z.B. Computer Literacy (ECDL) & Paradigmen von Programmiersprachen (Pizabeispiel)). Dieser Zustand bringt nichts besser zum Ausdruck als das immer noch andauernde Fehlen einer (möglichst internationalen) "Richtschnur" für informatische Bildung, wie sie in anderen Fächern bereits existiert. Kein methodisches Konzept hat bislang auf mich holistisch überzeugend gewirkt. - Zeitlich sind nahezu alle o.g. Inhaltspunkte im Sinne eines Spiralcurriculums über die Jahrgangsstufen verteilt (d.h. einer immer wiederkehrenden vertiefenden Wiederholung)

Aufgabe 2.)

Verständlicherweise erhebt auch die folgende Zusammenstellung von Methoden keinen Anspruch auf Vollständigkeit. Es ist damit zu rechnen, dass einige unterrichtliche Gegebenheiten u.U. nach anderen Methoden verlangen als diejenigen, in welchen ich momentan das größte Potential sehe. Im Sinne einer möglichst ausführlichen Beschreibung der von mir favorisierten Methoden (anstelle von einer holistischen Analyse meines Methodenfundus) seien an dieser Stelle drei Methoden näher betrachtet:

Puzzle-Methode:

Diese (mir bis vor kurzem noch vollkommen unbekannt Methode) schließt in meinen Augen eine nicht unwesentliche Lücke zwischen Gruppen- und Projektarbeit. Aktivität findet sowohl in Form von Einzelarbeit (Selbststudium) als auch in Kleingruppen statt und ist nahezu vollständig durch konkret vorgegebene Aufgaben gesteuert. Andererseits arbeitet die gesamte Lernergruppe aber auch gleichzeitig innerhalb eines großen gemeinsamen Themenbereichs (siehe Paradigmen von Programmiersprachen). Hervorragend zur Einführung in einen ganzen Themenkomplex geeignet, würde ich die Puzzle-Methode jedoch nicht zum vertieften Lernen innerhalb eines Zweigs des Themengebietes einsetzen.

Für diese ablehnende Haltung existieren verschiedene Gründe: Wesentliches Merkmal der Puzzle-Methode ist u.a. die zunächst getrennte Arbeit von „Expertengruppen“, die im Rahmen der Unterrichtsrunde ihr Wissen untereinander austauschen. Dies mag bei weit gefächerten Themen möglich sein, bei enggefassten Themen erscheint eine künstliche Aufspaltung jedoch überflüssig und darüber hinaus unnützlich. In diesem Fall müssen Methoden zum Einsatz kommen, die zwar ähnlich der Puzzle-Methode selbstverantwortliches Lernen betonen und schulen, jedoch nicht zwingend getrennte gleichzeitige Arbeit an verschiedenen(!) Perspektiven auf ein Problem verlangen.

Lerntheoretische Grundlagen:

- selbstverantwortliches Lernen begünstigt hohe (intrinsische!) Motivation
- Puzzle-Methode fordert / fördert Fähigkeit, Mitmenschen Aufmerksamkeit zu schenken und die eigenen Handlungen dagegen zu reflektieren
- durch eigenständige Erarbeitung der Grundlagen mittels Reader und Übungsaufgabe (Selbststudium) wird nicht deklaratives Wissen, sondern prozedurales Wissen aufgebaut

Projektarbeit:

Es ist eine häufig formulierte Annahme/Meinung, dass Informatikunterricht in besonderer Weise für Projektarbeit geeignet ist - dem möchte ich an dieser Stelle ausdrücklich zustimmen. De facto ist die gemeinsame Arbeit einer (häufig durchaus großen) Gruppe zur Lösung eines (vorgegebenen) Problems in der Softwareentwicklung der Regelfall. Vereinfacht lässt sich diese Form der Zusammenarbeit daher sicherlich auch in der schulischen informatischen Bildung umsetzen. Wichtig scheint mir an dieser Stelle vor allem klare Aufgabenverteilung, so dass der Versuch, Projektarbeitphasen zur Förderung des selbstbestimmten/-verantwortlichen Lernens einzusetzen, nicht einen eher gegenteiligen Effekt hat.

Lerntheoretische Grundlagen

- selbstverantwortliches Lernen begünstigt hohe (intrinsische!) Motivation
- Projektarbeit fordert / fördert ähnlich der Puzzle-Methode die Fähigkeit, Mitmenschen Aufmerksamkeit zu schenken und die eigenen Handlungen dagegen zu reflektieren
- Arbeit auf „gemeinsames“ Ziel hin begünstigt gegenseitige Unterstützung und eigene Zielstrebigkeit

Drills:

Dieser Begriff stammt eigentlich aus der Spracherwerbsforschung und bezeichnet das Einüben eines Prozesses „stur nach Schema F“. Obwohl die kreative individuelle Ausgestaltung durch den Lernenden bei Verwendung dieser Methodik nahezu ausgeschaltet ist, so gibt es doch Bereiche, in denen ein derartiges Vorgehen als durchaus sinnvoll erscheint. Sollen Lernende z.B. in die Lage versetzt werden, für ein gegebenes Zählproblem eine geeignete FOR-Schleife entwickeln zu können, so kann die Konstruktion von FOR-Schleifen in der Tat effizient mittels Drill-Verfahren eingeübt werden. - Zu beachten ist jedoch, dass vom Erfolg eines Drills (d.h. korrekter Output des Lernenden) in keinem Fall auf das Verständnis eines Sachverhalts geschlossen werden kann. Zwar ist es möglich, dass ein solches existiert, der korrekte Output jedoch ist lediglich ein Zeichen für die fehlerfreie Ausführung einer vorgegebenen Prozedur.

Lerntheoretische Grundlagen:

- Behaviorism
- ALM (Audiolingual Method)

Aufgabe 3.)

Konzeption eines subjektiv-idealen Informatikunterrichts

Beschreiben Sie das Verhältnis zwischen Lehrern und Schülern.

In der englischsprachigen Literatur zu SLA (Second Language Acquisition) und Fremdsprachendidaktik existiert ein Begriff, welcher ein m.E. ideales Verhältnis zwischen Lernenden und Lehrenden beschreibt: cooperative education. Dies meint insbesondere ein Verhalten des Lehrenden, das im wesentlichen durch folgende Aspekte gekennzeichnet ist:

- Lernende gestalten und bestimmen Unterrichtsablauf und Unterrichtsinhalt aktiv mit
- der Lehrende setzt dafür einen gewissen Rahmen, innerhalb dessen von den Lernenden selbstbestimmte Entscheidungen getroffen werden können
- grundsätzliche Annahme ist, dass der Lehrende nicht nach Nürnberger-Trichter-Modell "lehrt", sondern "Lernen" auf Schülerseite anregt
- gleichzeitig ist jedoch darauf zu achten, dass der Lehrende stets Autorität behält, die er nutzen kann/muss(!), um den Unterricht in notwendige Bahnen zu lenken bzw. Störungen zu unterbinden
- **WESENTLICHES KENNTZEICHEN DES MODELLS:** Alle Interaktionen von Lehrenden und Lernenden sind durch Gegenseitigkeit gekennzeichnet. D.h. es wird davon ausgegangen, dass sowohl Schüler als auch Lehrer im Unterricht "lernen"; der Lehrende soll dabei jederzeit Hilfestellung geben, sofern diese benötigt wird. Gleichzeitig ist es seine Aufgabe, den Unterricht unter Berücksichtigung von Schülerinteressen und curricularen Gesichtspunkten zu lenken (d.h. ihm eine klar erkennbare Konzeption bzw. Richtung zu geben, die einer didaktischen Konzeption stand-hält)

Meiner Meinung nach lässt sich dieses Konzept geradezu ideal auf den Informatikunterricht übertragen. Insbesondere die Aspekte der Gegenseitigkeit und der Schülerinteressen treten in diesem Zusammenhang in den Vordergrund u.a. war letzterer Punkt Thema des ersten Übungszettels. Es gilt somit, Schnittstellen zwischen Schülerinteressen und Curriculum zu finden, so dass grundlegendes Wissen (Prinzipien und Methoden) durch von den Lernenden motivierte Themen/Anwendungsbeispiele vermittelt werden

kann. Explizit sollte darauf hingewiesen werden, dass dies NICHT eine ausschließliche Orientierung an Lernerinteressen kennzeichnet; letztere fungieren hier als "Transportmittel" für grundlegende Prinzipien.

Welchen Stellenwert hat ihre Halbjahresplanung, Planung der Unterrichtsreihe und Planung der Unterrichtsstunde?

Zusammenfassend lässt sich vorweg hervorheben, dass die Abstraktion mit der Länge des zu planenden Zeitraums wachsen sollte. D.h. eine Halbjahresplanung sollte vorsehen, welche Unterrichtsreihen (lediglich an Prinzipien und Methoden orientiert keine konkrete Ausarbeitung) zu welchem Zeitpunkt eines Halbjahres stattfinden sollen. Konkrete Unterrichtsreihen können erst geplant werden, wenn Informationen über Schülerinteressen vorliegen zwar sollte von vornherein feststehen, auf welche grundlegenden Konzepte eine Unterrichtsreihe abzielen, die konkrete Ausarbeitung sollte aber die Interessen der Lernenden als "Kanal" verwenden, durch den Fundamentales vermittelt wird. Die Planung einer Unterrichtsstunde schließlich sollte so konkret wie möglich gestaltet werden jedoch ist auch hier zwingend auf ausreichenden Raum für Kreativität und Spontaneität zu achten, da eine Einbeziehung jedweder unterrichtlicher Eventualität unmöglich scheint (es stellt sich an dieser Stelle auch die Frage, ob Unterricht, der lediglich rigide einer Planung folgt, nicht eher an Reiz verliert als gewinnt). Was jedoch sagen diese Abstraktionsgrade nun über den Stellenwert der verschiedenen Planungen im Rahmen meines Informatikunterrichts aus?

Es ist äußerst schwierig, den genannten Elementen der Halbjahresplanung, Planung der Unterrichtsreihe und Planung der Unterrichtsstunde definitive (und womöglich verschiedene Stellenwerte) zuzuordnen. Alle genannten Planungen haben ihren Platz, Sinn und Zweck und sollten damit auch gleich gewichtet werden. Entscheidend ist nicht ihr Stellenwert, sondern vielmehr der Raum für Ausgestaltung, welcher durch die einzelnen Planungsschritte offengehalten bzw. geschlossen wird.

In welchem Verhältnis stehen die Inhalte zur Methode?

An dieser Stelle möchte ich einleitend auf ein Beispiel von Prof. Jürgen Vorsmann verweisen, seines Zeichens emeritierter Univesitätsprofessor an der Universität Dortmund im Bereich Unterrichtsplanung und Didaktik (E2/E4) des Fachbereichs 12 (Erziehungswissenschaften). Dieser beschrieb eine (fiktive) Unterrichtssituation, in welcher ein Lehrer darum bemüht ist, seinen Schülerinnen und Schülern das Konzept "Demokratie" näherzubringen; dies geschieht in der Form eines Lehrervortrags. Wiederholt heben Schüler dabei die Hände, um den Wunsch nach mündlicher Beteiligung zum Ausdruck zu bringen. Leider werden in der Beispielsituation jegliche Versuche dieser Art augenblicklich durch den Lehrenden unterbunden, etwa durch Formulierungen wie: "Jetzt lasst mich doch mal ausreden!" o.ä.. Was ist nun das Besondere an dem geschilderten Verhalten des Lehrers?

Auch für den Laien ist sicherlich zumindest eins zu erkennen: Das Verhalten des Lehrers ist vollkommen konträr zum eigentlichen Unterrichtsinhalt! Demokratie als Konzept enthält als Kennzeichnendes Element das "Anhören" auch von Einzelstimmen und anderen Meinungen ("gesellschaftliche Mitbestimmung"). Dem Vorgehen des Lehrenden jedoch gehen exakt diese Elemente ab ein Zustand der aus diversen Gründen nicht förderlich scheint: In erster Konsequenz erfahren die Lernenden am eigenen Leib, dass so gut die Idee von Demokratie auch sein mag, eine Umsetzung allerdings nicht stattfindet (und somit evtl. utopisch erscheint). In zweiter Konsequenz schließlich erscheint es nicht unwahrscheinlich, dass die Schülerinnen und Schüler Demokratie (evtl. unbewusst) als unrealistisch verwerfen und niemals in der Lage sind, das in diesem Konzept enthaltene Potential in voller Gänze wahrzunehmen. Obwohl Informatikunterricht sicherlich nicht primär "Demokratie" per se als Thema behandelt (obwohl sich durchaus Schnittstellen mit dem Themenkomplex der informatischen Selbstbestimmung aufzeigen lassen), weist die geschilderte Situation doch einige Charakteristika auf, die wertvolle Hinweise auf das Verhältnis zwischen Methoden und Inhalten des Informatikunterrichts zu liefern in der Lage sind.

Vereinfacht kann gesagt werden, dass Methode und Inhalt korrespondieren müssen. Verschiedene Sichtweisen auf ein und das selbe Problem (wie etwa beim Pizzabeispiel) sind nicht unbedingt ideal per fragend-entwickelndem Unterrichtsgespräch zu vermitteln. Vielmehr muss hier im Vordergrund stehen, dass die Lernenden weitestgehend selbständig ihr Wissen aktiv erweitern. Im Gegensatz dazu ist aber der Prinzipielle Aufbau einer FOR-Schleife eher vom Lehrer an die Schüler zu vermitteln hier geht es zunächst eher um Faktenwissen selbst als um dessen Anwendung. In einem zweiten Schritt kann die Einarbeitung in die Benutzung des FOR-Konstruktes wieder unter aktiver Beteiligung der Lernenden geschehen. Zusätzlich zu den bisher getroffenen Aussagen räumen diese Erkenntnisse ebenfalls mit einem weit verbreiteten Vorurteil bzgl. der schlechten Eignung von Frontalunterricht als Lehrmethode auf. Sicherlich sind als "Lehrervortrag" gestaltete Unterrichtsstunden absolut indiskutabel, punktuell jedoch scheint eine knappe Frontalphase durchaus angebracht, sofern sie sich methodisch und inhaltlich in das Geschehen einer Unterrichtsstunde fügen lässt.

In welcher Form findet Leistungsbewertung statt? Wodurch unterscheidet sich eine "gute", eine "befriedigende" und eine "ausreichende" Leistung der Schüler?

Hinweis:

Zur Bearbeitung dieses Teilaspektes werde ich Leistungsbewertung als von mir frei bestimmbar betrachten (ohne Einschränkung durch etwaige Richtlinien) schließlich soll Informatikunterricht beschrieben werden, der meinen Vorstellungen entsprechend einen hohen Grad an Idealität besitzt. Im Idealfall erstreckt sich die Bewertung von Schülerleistung über das ganze Schulhalbjahr und setzt sich aus verschiedenen Elementen zusammen. So ist z.B. eine ähnlich geartete Aufschlüsselung der Endnote denkbar, wie sie beim Deutschunterricht der University of Iowa zu Anwendung kommt (selbstverständlich nach entsprechender Anpassung der einzelnen Elemente):

| | |
|--|---|
| <i>"Attendance & Participation</i> | <i>15%</i> |
| <i>Homework</i> | <i>20%</i> |
| <i>Cultural Exploration</i> | <i>10%</i> |
| <i>Quizzes</i> | <i>10%</i> |
| <i>Chapter Tests</i> | <i>15% [Klassenarbeiten bzw. Klausuren]</i> |
| <i>Final Exam</i> | <i>30%</i> |

Ein derartiges Bewertungsmodell befreit auch Informatikunterricht (wie jedweden anderen Unterricht) von der in Deutschland häufig anzutreffenden "Allmacht der Klausuren / Klassenarbeiten". Leistungsbewertung findet somit konsequent das gesamte Halbjahr hindurch statt, was nicht nur Prüfungsangst und punktuellen Leistungsdruck vor Klausuren reduziert, sondern darüber hinaus gewiss einen exakteren Blick auf die Leistungsstände der Lernenden ermöglicht als ein mehr oder minder rein klausurdominiertes Modell. Gerade das Fach Informatik ist durch seine (bekannte) Eignung für Projektunterrichtsphasen prädestiniert, eine kontinuierliche Bewertung anstelle von punktueller Leistungsmessung zur Anwendung zu bringen.

Zur in der Aufgabenstellung angeführten Klassifizierung von Leistungen in "gut", "befriedigend" & "ausreichend" sollte zu jedem Zeitpunkt der individuelle Schüler / die individuelle Schülerin im Mittelpunkt stehen. D.h. eine Leistung ist nicht nur dann "gut", wenn sie unter Aspekten wie Intelligenz "genial" ist, sondern wenn sie eigene, kreative, ambitionierte (und selbstverständlich auch größtenteils korrekte) Auseinandersetzung mit dem Unterrichtsinhalt erkennen lässt. Die von mir verwendete Formulierung lässt darauf schließen, dass ich ernsthafte Bemühung und Korrektheit einer Lösung/Leistung nahezu gleichgewichte. Wie sonst kann den teils enorm unterschiedlichen Voraussetzungen innerhalb einer Lerngruppe begegnet werden? Ausdrücklich soll darauf hingewiesen werden, dass dieser Ansatz ist nicht mit jener Methodik übereinstimmt, die von Schülerinnen und Schülern mit guten Vorkenntnissen per se mehr Leistung in Form von Mehrarbeit verlangen. Viel mehr betont Sie die Notwendigkeit des Vorhandenseins einer Bereitschaft zum aktiven Lernen auf Schülerseite. Eine "befriedigende" Leistung weist

konsequenterweise deutliche Mängel in einem der beiden genannten Bereiche (Korrektheit u. "Lernebereitschaft" / Motivation) auf, einer „ausreichende“ fehlt es an beidem.

Inwieweit lässt sich ihr „Modell“ auf den Unterricht Ihrer (vorgestellten) Fachkollegen übertragen?

Die Frage ist m.E. eher weniger, ob sich mein Modell auf den Unterricht anderer Lehrender übertragen lässt... davon gehe ich aus; schließlich ist das Modell zwangsläufig vor dem Hintergrund einer potentiellen Umsetzbarkeit entstanden (ansonsten wäre ein Modell unrealistisch). Somit ist die prinzipielle Übertragbarkeit sicher. Viel interessanter ist aber vielmehr, ob sich nicht Widerstand auf Seiten meiner Fachkollegen zeigen würde.

Letzteres wäre nicht nur verständlich, es ist zu erwarten! Es sollte nicht vergessen werden, dass es sich bei der von mir dargelegten Konzeption um meine Idealvorstellung handelt, d.h. es ist ein Modell, welches meiner Meinung nach bestmöglich mit meiner Persönlichkeit und meinen Ansichten korrespondiert. Dass diese bei Fachkollegen teils oder in umfangreicher Weise abweichen und daher meine Vorstellung von Informatikunterricht auf Widerstand stößt, kann als gegeben angenommen werden.

Übungsblatt 12

Dieses abschließende Übungsblatt der Lehrveranstaltung „Didaktik der Informatik III (für Sek I)“ soll Ihnen die Gelegenheit bieten, Stellung zu beziehen zu ihrer persönlichen Vorstellung eines „guten Informatikunterrichts“.

Aufgabe 1:

Welche Inhalte halten Sie persönlich für die Sekundarstufe I für wesentlich? Differenzieren Sie hierbei nach unterschiedlichen Altersstufen. Ordnen Sie darüber hinaus die genannten Teilgebiete den Fachgebieten zu, wie Sie sie z.B. aus der (universitären) Fachsystematik kennen.

Im Folgenden werden - rein subjektiv - die Vorstellungen des Autors bezüglich eines guten Informatikunterrichts in der Sek 1 wiedergegeben. Diese gehen dabei strukturell über die gegenwärtigen Gegebenheiten hinaus. Beispielsweise gehe ich von einem durchgehenden vierstündigen Informatikunterricht von Beginn der Jahrgangsstufe 6 aus. Hierbei werden folgende Grundsätze angewendet: Da ich grundsätzlich gegen einen Paradigmenwechsel (in beiden Sekundarstufen) bin, weil ich der (subjektiven) Ansicht bin, dass den Paradigmen „Logik“ und „Funktional“ fachlich weniger Bedeutung zukommt als dem objektorientierten Paradigma, würde ich auf diese Inhalte verzichten. Zudem erübrigt sich das imperative Paradigma da es bereits im objektorientierten enthalten ist.

Die Stoffüberladung eines Faches führt in seltenen Stellen zu wirklichem Verständnis, sondern häufig eher zu Verwirrung, wenn die Paradigmen zu hastig eingeführt werden. Strukturell ist eine Kooperation mit anderen Fächern durchaus wünschenswert, allerdings ist hier insbesondere beim Fach Mathematik (obwohl oder gerade weil dies mein anderes Fach ist) Vorsicht geboten, da eine Kooperation mit der Mathematik dazu neigt, Schüler abzuschrecken, mathematische Inhalte (wie z.B. Algorithmen wie den Euklid. Alg.) in die Informatik zu verlagern und bei den Schülern zudem den Eindruck erzeugen kann, dass sich viele Probleme mit informatischen Methoden einfacher lösen lassen als mit mathematischen und die Mathematik somit überflüssig sei. Die Fächer sollten also getrennt voneinander existieren. Inhaltlich würde ich in der 6./7. Klasse also zunächst mit der Bedienung der Informatiksysteme beginnen. Dies kann mit verschiedenen Inhalten unterlegt werden. Das Erstellen von Dokumenten, Bildern und die Kommunikation im Netz über E-Mail sind Inhalte an denen dies faktisch geschehen kann. Um in diesem Teil nicht nur Produktschulung zu betreiben, würde ich hier schon mit der Motivation erster objektorientierter Begriffe beginnen (wie in [Hub01] ausgeführt). Diese können beispielsweise anhand eines Malprogrammes stattfinden und so den Objektbegriff und auch den Klassenbegriff einführen. Dieser wird später im Kontext der Programmierung von Bedeutung sein. Im Rahmen der Fachsystematik findet sich dies in der objektorientierten Modellierung wieder (also Vorlesung „Datenstrukturen, Algorithmen und Programmierung I“). Ferner würde ich die Repräsentation von Information über die Begriffe Vektorgraphik vs. Pixelgraphik motivieren und somit die Grundlage für die Vorstellung eines Datentyps geben). Außerdem würde ich das von-Neumann-Addierwerk durchnehmen und damit den prinzipiellen Aufbau eines Rechners. (Auf Wunsch der Schüler sind „Ausflüge“ in eher praxisnähere Probleme denkbar.).

Daran anschließend würde ich in der 8./9. Klasse die Programmierung in den Mittelpunkt der Aufmerksamkeit rücken. Zunächst würden einfache imperative Programme im Mittelpunkt stehen. Ein wichtiger Punkt ist dann der Übergang zur Objektorientierung, wobei auf die vorher erlernten Konzepte zurückgegriffen wird. Die Programmierung dominiert in weiten Teilen die 8. und 9. Klasse. (fachlich ist dies die Programmierung, also ebenfalls DAP1).

Ein wichtiges Thema, für das auf jeden Fall Zeit aufgewendet werden muss, ist das Thema „Daten- und Persönlichkeitsschutz“, dem immer mehr Bedeutung zukommt und für die Schüler vermutlich von grosser Lebensrelevanz sein wird. In diesem Zusammenhang ist auch die Diskussion ethischer Probleme im Zusammenhang mit Informatiksystemen relevant (fachlich ist dies der Bereich „Informatik und Gesellschaft“). Dies sollte an das bestehende Fachwissen der Schüler durch nähere Behandlung des

Themas „Kommunikation in Netzwerken“ verstärkt werden. (Fachlich: „Rechnernetze und verteilte Systeme“).

In der 10. Klasse würde zunächst das Automatenmodell eingeführt werden (beispiel: Colaautomat), was dann mittels der Programmierkenntnisse umgesetzt werden kann. Hier sollen den Schülern die Grenzen eines Modells (hier des Automatenmodells) anschaulich vermittelt werden. In der 10.2 folgt ein größeres Projekt im Rahmen der Programmierung. Hierbei sollen die zuvor erworbenen Fachkenntnisse jedoch nicht nur wiederholt, sondern zudem maßgeblich ergänzt werden durch die Erkenntnis, das ein Programmierprojekt einen nicht geringen Verwaltungsaufwand besitzt. (fachlich findet dieser Teil in der Softwaretechnik wieder).

(Diese Planung ist eine Minimalplanung, näheres dazu in Aufgabe 3).

Aufgabe 2:

Welche Methoden sollten in Ihrem Informatikunterricht zum Einsatz kommen? Zeigen Sie die lerntheoretischen Grundlagen Ihrer bevorzugten Methoden auf und beschreiben Sie, in welchem Umfang sie welche Methode einsetzen möchten. Stellen Sie dabei ggf. auch Bezüge zu den in Aufgabe 1 genannten Inhalten her.

Der Begriff „Unterrichtsmethode“ ist ein recht weiter Begriff, der i.A. näher ausdifferenziert wird in „Handlungsmuster“ bzw. „Sozialformen“, manchmal auch in „Unterrichtsschritte“, „Handlungsmuster“ und „Methodische Grossformen“ untergliedert. Die Sozialformen des Unterrichts werden hierbei unterteilt in „Frontalunterricht“, „Gruppenunterricht“, „Partnerarbeit“ und „Einzelarbeit“. Typischerweise ist der Frontalunterricht die am Meisten verbreitete Sozialform. (um die 80% lt. [Mey87]), „Gruppenunterricht“ oder „Partnerarbeit“ treten dem gegenüber quantitativ zumeist in den Hintergrund. Den Frontalunterricht unter den weniger bevorzugten Methoden aufzuführen, halte ich deshalb auch für unrealistisch, da ihm in der Schulpraxis aus technischen Gründen (Lehrer-Schüler-Relation, Raumknappheit, etc.) häufig die Funktion zukommt, Sach- und Problemzusammenhänge zu erarbeiten. Eine lerntheoretische Fundierung dieser Methode ist mir auch nicht bekannt, angesichts der körperlichen Bewegungsarmut, der passiven Grundhaltung (vermutlich kommt ein behavioristischer Ansatz hier der Idee am nächsten) ist die Effektivität allgemein anzuzweifeln. Frontalunterricht hat jedoch den Vorteil, dass es eine Sozialform ist, die dem Schüler bekannt ist und er den Grad seiner Aktivität aktiv mitbestimmen kann. Häufig wird dem Frontalunterricht der Gruppenunterricht gegenübergestellt, der verschiedene Ausprägungen (z.B. im Informatikprojekt) haben kann und mit etwa 5-7% eher weniger genutzt wird. Anders als der Frontalunterricht lässt sich der Gruppenunterricht durch theoretische Überlegungen begründen (Entwicklung von Rollendistanz, Empathie, Frustrationstoleranz, solidarischem Handeln). Ich persönlich halte dies für eine recht anspruchsvolle Unterrichtsform, die jedoch sehr grosse Kompetenzen von Seiten des Lehrers wie der Schüler verlangt, bei den Schülern häufig mit geringen oder sogar negativen Vorerfahren verbunden ist und deshalb evtl. Gefahr läuft, die entsprechenden Ziele nicht zu erreichen. Der Partnerarbeit ist die am wenigsten verwendete Sozialform, ihr kommt in der Informatik m.E. eine verstärkte Bedeutung zu. Ich halte dies für einen Ausgangspunkt in Richtung Gruppenarbeit. Abseits von Sozialformen, halte ich insbesondere eine große Handlungsorientierung (z.B. bei Suchalgorithmen oder bei dem Begriff der Rekursion) oder ein Rollenspiel für sinnvoll, sich dadurch motivieren lässt, daß der Lehrer einen Gegenstand in den Unterricht mitbringt, was typischerweise für Aufsehen sorgt und einen (hoffentlich) interessanten Unterrichtseinstieg bietet.

Aufgabe 3:

Stellen Sie eine Konzeption Ihres Informatikunterrichts dar. Beschreiben Sie das Verhältnis zwischen Lehrern und Schülern. Welchen Stellenwert hat Ihre Halbjahresplanung, Planung der Unterrichtsreihe und Planung der Unterrichtsstunde? In welchem Verhältnis stehen die Inhalte zur Methode? In welcher Form findet Leistungsbewertung statt? Wodurch unterscheidet sich eine „gute“, eine „befriedigende“ oder „ausreichende“ Leistung der Schüler? Inwieweit lässt sich Ihr „Modell“ auf den Unterricht ihrer vorgestellten Fachkollegen übertragen?

Grundsätzlich halte ich es für empfehlenswert, die Planung als eine Minimalplanung darzustellen, d.h. die Planung enthält Themen, die definitiv behandelt werden und darüber hinaus fakultative Themen, die definitiv nicht alle behandelt werden. Hierdurch soll eine Überladung des Stundenplans verhindert werden. Da durch den typischerweise vorkommenden Stundenausfall viele Inhalte wegfallen, entsteht so ein Skelett, das (hoffentlich) durch Inhalte aus dem fakultativen Bereich ergänzt werden kann. Hierbei ist es möglich, mit den Schülern quasi in „Verhandlung“ zu treten und so das Skelett durch für Schüler interessante Themen zu ergänzen. So erhalten die Schüler auch die Möglichkeit, eigene Ideen einzubringen und auf die explizite Ausführung Einfluss zu nehmen. Das „Skelett“ dient hierbei dazu, Schülern wie Lehrern eine gewisse Sicherheit zu geben (Schüler haben ja durchaus auch ein Interesse daran, bei einem Lehrerwechsel einen bestimmten Standardinhalt gehabt zu haben). Die gewählte Methode hängt hierbei in starkem Masse von dem Inhalt ab, allerdings trifft auch die umgekehrte Beziehung zu. In eher informativen Phasen (oder Phasen, die informativ sein sollen) tritt der Frontalunterricht verstärkt auf, dem zwar eindeutig auch im Kontext des Informatikunterricht die größte Bedeutung zukommen soll, allerdings nicht in dem Maße (über 80%), wie dies typischerweise gegenwärtig der Fall ist. In eher meinungsabhängigen Phasen bietet sich die Diskussion an, für Übungen am Rechner die Partnerarbeit und für Projekt der Gruppenunterricht, der allerdings - bei allen Vorteilen - dem Lehrer eine besondere Aufmerksamkeit abverlangt.

Die Leistungsbewertung findet zum einen durch Klassenarbeiten statt, zum anderen durch die sonstige Mitarbeit, zu der allerdings nicht nur die mündliche Mitarbeit zählt, sondern - wie in den Lehrplänen angegeben - auch die anderen Zeichen der Mitarbeit, wie die Hausaufgabe, die Heftführung, evtl. ein Referat oder Protokoll, eine schriftliche Übung (Test), Mitarbeit in einem Team (z.B. bei einem Informatikprojekt) sowie der Umgang mit bestehenden Informatiksystemen, also den Schulrechnern z.B. durch die Unterstützung anderer Schüler im Umgang mit den Rechnern sind im Informatikunterricht von großer Bedeutung und ermöglichen es, die mündliche Mitarbeit, die m.E. häufig dazu herangezogen wird, einen sehr subjektiven Eindruck des Lehrers vom Unterrichtsgeschehen in der Notenfindung wiederzugeben, durch objektivere Kriterien abzulösen. Zudem erhalten Schüler, die sich ungern spontan am Unterrichtsgespräch beteiligen die Gelegenheit, sich in geeigneter Form in den Unterricht einzubringen.

In Bezug auf die Leistungsbewertung würde ich folgende Kriterien (die Teilweise aus den KMK-Dokumenten bzgl. der Sek. II (!) abgeleitet sind [Hrs91]). Dieses teilt die Leistungen in Reproduktion, Reorganisation und Transfer ein, wobei allerdings schwer zu definieren ist, wo die Grenze zu ziehen ist. Meiner Ansicht nach ist dies auch nur schwer objektivierbar. In einer Arbeit würden demzufolge die produktiven Leistungen unter 40% liegen, die Reorganisation zwischen 30% und 40% und der Transfer somit zwischen 15% und 30%. Nach diesen Vorgaben sollte einer Note ausreichend bei 40% der Gesamtleistung liegen, eine gute Klausur zwischen 70% und 85%. Für eine sehr gute Leistung sind Transferleistungen erforderlich. Die Übertragung dieser Richtlinien in die Sekundarstufe I ist nicht problemlos möglich, zumal hier dem Transfer eine etwas geringere Bedeutung zukommt. Bei einer geeigneten Definition der Begriffe Reproduktion, Reorganisation und Transfer (die eigentlich nur im Unterrichtskontext möglich erscheint), halte ich eine Übertragung in die Sek. I allerdings für möglich. (wobei dem Transfer eine deutlich geringere Bedeutung zukommt). Von großer Relevanz ist auch, dass die Note sich nach dem Bearbeitungsaufwand und nicht nach dem Schwierigkeitsgrad richtet. Dies sollte den Schülern allerdings auch mitgeteilt werden.

In Bezug auf die sonstige Mitarbeit, würde ich folgende Einteilung für sinnvoll halten:

| | |
|--|-----|
| Mündliche Mitarbeit | 20% |
| Hausaufgaben | 25% |
| Heftführung | 15% |
| Test | 20% |
| Mitarbeit in einem Team | 10% |
| Umgang mit Informatiksystemen der Schule | 10% |

Ich interpretiere den letzten Teil dieser Aufgabe dahingehend, dass der Zusammenhang meines Unterrichts mit dem anderer (virtueller) Informatiklehrer zu diskutieren ist. Durch die Festlegung minimaler Inhalte, die in jedem Fall behandelt werden, wird eine Übertragung möglich. Problematisch kann die Einführung objektorientierter Prinzipien sein, die auf den Ergebnissen der 6. und 7. Jahrgangsstufe aufbauen.

Literatur

- [Hrs91] Hrsg, KMK: Einheitliche Prüfungsanforderungen in der Abiturprüfung „Informatik“. 1. Luchterhand : Neuwied, 1991
- [Hub01] Hubwieser, Peter: Didaktik der Informatik: Grundlagen, Konzepte, Beispiele. 1. Berlin : Springer, 2001
- [Mey87] Meyer, Hilbert: Unterrichtsmethoden II: Praxisband. 13. Berlin : Cornelsen Verlag Scriptor GmbH & Co. KG, 1987

Übungsblatt 12

Freiheit, Ungleichheit, Fairness, Respekt und Würde

Aufgabe 3

Jeder Unterricht sollte diese fünf Grundsätze berücksichtigen. Dies betrifft Lehrkräfte, Schülerinnen und Schüler, Methoden und Themen/Inhalte/Gegenstände.

Konkret bedeutet das zum Beispiel, dass alle Lehrkräfte immer wieder eine lang-, mittel- und kurzfristige Planung machen (Würde und Respekt gegenüber den Schülerinnen und Schülern). Die Planung trägt den Ungleichheiten der Lerngruppe Rechnung und ist (fairerweise) nicht darauf ausgerichtet, eine gewisse Durchfallquote zu erreichen.

Traditionell gedacht ist die Planung zudem flexibel oder kann geändert werden, um Wünsche und Anregungen der Schülerinnen und Schüler zu berücksichtigen – ob dies in einem vollständig demokratischen Modus passiert, ist eine andere Frage (Freiheit der Lehrkräfte und der Lerngruppe). Nicht traditionell gedacht entscheiden die Schülerinnen und Schüler selbst, ob sie an der geplanten Veranstaltung teilhaben möchten.

Die Leistungsbewertung ordnet sich ebenfalls den Grundsätzen unter. Freiheit: Die Schülerinnen und Schüler wählen Termine von Prüfungen im Rahmen des Schulhalbjahres selbst. Ungleichheit: Es gibt ein differenziertes Notenspektrum, das den aktuellen Leistungsstand widerspiegelt. Fairness: Modalitäten der Leistungsbewertung sind der Lerngruppe im Vorhinein bekannt – auch für „mündliche“ Noten! Respekt und Würde: Individuelle Stärken werden positiv berücksichtigt und Schwächen nicht bestraft (zumindest nicht als Strafe empfunden).

Wie könnte eine solche Leistungsbewertung – sogar in der heutigen Schule – aussehen?

Es gibt feste, regelmäßige Termine für Prüfungen; die Schülerinnen und Schüler stellen ihre persönliche Reihenfolge auf und können in einzelnen Fächern zum gleichen Thema auch mehrfach teilnehmen, wobei dann die bessere Note gewertet wird. Die Definitionen von guten, befriedigenden und ausreichenden Leistungen können aus den bestehenden Vorschriften übernommen werden – eine befriedigende Leistung müsste allerdings im Vorhinein definiert werden. Für die Betonung von Stärken können mehrere Mittelwerte gebildet und der beste für die Zeugnisnote verwendet werden. Solche Mittelwerte können zum Beispiel mündliche und schriftliche Leistungen verschieden gewichten oder aus einem Spektrum vorhandener Noten nur einen Anteil verwenden (wie es in machen Sportarten üblich ist). – Nach meiner Erfahrung wird eine solche Verstärkung im Sinne der Psychologie bei der Notengebung in der Schule sträflich vernachlässigt. Das Verhältnis zwischen Lehrkräften und den Schülerinnen und Schülern ist ebenfalls von den Grundsätzen geprägt. Sie geben/lassen sich gegenseitig Freiheiten, wissen, dass sie in der Institution Schule unterschiedliche Positionen haben, und ihr Verhalten ist (trotzdem?) durch Fairness, Respekt und Würde geleitet – das verlange ich von jedem; es gibt Autoren, die an dieser Stelle viel weiter gehen.

Ebenso verhält es sich mit Inhalt und Methode – Respekt und Würde fallen hier heraus.

Themen/Inhalte/Gegenstände sollten nicht an Methoden gebunden sein, sondern durch möglichst ungleiche, sogar gegensätzliche methodische Ansätze erfahrbar sein.

Fairness ist (a) durch Integration verschiedener Ansätze bei gleichem Inhalt und (b) durch Variation der Inhalte bei gleicher Methode zu gewährleisten. Präziser wird dies nur gewährleistet, wenn den Schülerinnen und Schülern (a) verschiedene Betrachtungswinkel und (b) universelle Methoden klar werden. (Fach-) Kollegen sollten das als Ganzes nicht einfach übertragen, sondern müssen selbst herausfinden, was zu ihnen passt und was nicht. Es geht schließlich um ein freies, vielfältiges, faires, respektvolles und würdiges Miteinander.

Aufgabe 1

Die großen Bereiche sind schnell abgesteckt: Informatiksysteme anwenden, modellieren, bewerten und die Kombinationen dieser Tätigkeiten. Im Zusammenhang mit konkreteren Inhalten ist die Lerngruppe

der entscheidende Faktor. Es macht schließlich einen Unterschied aus, ob eine siebte Klasse unbefangen zum ersten Informatikunterricht kommt oder in der fünften und sechsten Stufe schon welchen erlebt hat und bei wem. Es macht einen Unterschied aus, ob die Klasse in der Lage ist oder nicht, inhaltliche Anregungen und Wünsche zu äußern oder sich selbst zu bewerten. Daher weigere ich mich schlicht, konkrete Inhalte anzugeben – die Alternative „Vision eines Lehrplanes“ für die Informatik in der Sekundarstufe I sprengt den Rahmen dieser Aufgabe.

Aufgabe 2

Beim Begriff „Methoden“ betreten wir nun endgültig den Sumpf der Didaktik. Die zwei wichtigsten Wege zu Erkenntnis und Wissen – induktiv und deduktiv – werden fast nie im Rahmen von Unterrichtsmethoden genannt. Viel mehr ist von „Sozialformen“ (Frontalunterricht, Gruppenunterricht, Partnerarbeit, Einzelarbeit) und „Handlungsmustern“ (es gibt zu viele, um sie hier kurz aufzulisten) oder Projektarbeit, Handlungsorientierung, etc. die Rede.

In der Lerntheorie gibt es drei Paradigmen: Behaviorismus, Kognitivismus und Konstruktivismus.

- Behaviorismus: Lernen ist das Herstellen einer gewünschten Input/Output-Relation (Stimuli/Reiz und Reaktion). Dazu gehören das klassische Konditionieren und das operante Konditionieren (die in Aufgabe 3 angerissene Verstärkungstheorie).
- Kognitivismus: Lernen besteht aus dem Aneignen von Methoden. Dazu gehört das soziale Lernen. Synonyme aus der Lerntheorie sind Modelllernen und sozial-kognitives Lernen. Darunter ist also nicht das soziale Lernen zu verstehen, wie es zum Beispiel im Lehrplan Informatik für die Sekundarstufe II (Seite 72?) Verwendung findet – dort wird es gleichgesetzt mit dem Lernen prosozialen Verhaltens. Zum Kognitivismus gehört ferner das Lernen durch Einsicht (Informationsverarbeitungsprozess im Gehirn hat den richtigen Weg gefunden und löst den Aha-Effekt aus, wenn die Schüler und Schülerinnen plötzlich zu verstehen glauben, was die Lehrkraft oder Mitschülerinnen und -schüler zu erklären versuchen).
- Konstruktivismus: Beim Lernen geht es um das Bewältigen komplexer Situationen – aus der Pädagogischen Psychologie kenne ich dazu nichts weiter.

Einzelarbeit gehört im Sinne eines Trainings (Konditionierung) zum Behaviorismus, Frontalunterricht, Partnerarbeit, Gruppenunterricht und Projektarbeit zum Kognitivismus und Handlungsorientierung zum Konstruktivismus – manche Kombination wie handlungsorientierte Projektarbeit ist natürlich möglich. Keine dieser Methoden kann ich bevorzugt nennen, denn sie sind für manches wichtiger als für anderes: Mausbedienung lernt man nicht im Gruppenunterricht, sondern durch Training, für die alleinige Aneignung von Fakten ist Frontalunterricht besser als Projektarbeit (letzteres kann aber Anlass dazu sein) und Demokratie erfährt man nicht im Frontalunterricht, sondern durch Projektarbeit und Handlungsorientierung. Die einzig bevorzugte Methode die induktive Methode, mit der wir von Kindesbeinen an sehr erfolgreich lernen. Die Deduktion kann zur Abgrenzung und Verdeutlichung von Grenzfällen dienen, die induktiv nur durch sehr viele Beispiele erfassbar wären.

Endnote

(*) Mein besonderer Dank gilt Martin Reinertz für seine vollständigen und jeweils sehr ausführlichen Abgaben. Sie sind in diesem Dokument mit einem (*) gekennzeichnet.
Athanasios Papoulias