



Übung zur Vorlesung „Didaktik der Informatik II“

Blatt 1

In der Vorlesung wurde Bezug genommen auf einen aktuellen Beitrag von Monika Seiffert¹, der einen Übergang des Gesamtkonzepts informatischer Bildung der GI² zum Informatikunterricht in der Sekundarstufe II aufzeigt.

1. Aufgabe: (0 Punkte)

Lesen Sie den Beitrag.

2. Aufgabe: (2 Punkte)

Zeigen Sie auf, welchen Beitrag zur Allgemeinbildung der Informatikunterricht zu leisten vermag.

3. Aufgabe: (2 Punkte)

Stellen Sie heraus, welche Schwierigkeiten entstehen können, wenn die Struktur der Fachwissenschaft Informatik in den Unterricht übertragen wird. Nennen Sie Alternativen.

4. Aufgabe: (4 Punkte)

Geben Sie mit eigenen Worten wieder, wie das vorgestellte Gesamtcurriculum (für die Sekundarstufe I und II) aufgebaut ist.

5. Aufgabe: (2 Punkte)

Da sich der Informatikunterricht in der Sekundarstufe I im Wahlbereich befindet, können Schülerinnen und Schüler beim Eintritt in die Sekundarstufe II unterschiedliche Vorkenntnisse haben. Wie gehen Sie mit dieser Situation um?

Abgabe: 23.10.2003

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ [Seiffert 2003] SEIFFERT, Monika: Vom Gesamtkonzept zum Curriculum. Planung von Kurssequenzen. In: LOG IN (2003), Nr. 124, S. S. 10–16

² [GI 2000] GI: Empfehlung der Gesellschaft für Informatik e.V. für ein Gesamtkonzept zur informatischen Bildung an allgemein bildenden Schulen. In: Informatik Spektrum 23 (2000), Dezember, Nr. 6, S. 378–382. – http://www.gi-ev.de/informatik/publikationen/gesamtkonzept_26_9_2000.pdf – geprüft: 15. Oktober 2003

Übung Blatt 1 DDI II
XXX

2.Aufgabe:

In Anlehnung an den vorliegenden Artikel¹ können folgende Beiträge des Informatikunterrichts zur Allgemeinbildung benannt werden:

- Durch das Strukturieren und Modellieren von Zusammenhängen und Problemen wird die Abstraktionsfähigkeit und das Problemlösen trainiert. Damit erhalten die Schüler ein Werkzeug dass nicht auf den Bereich der Informatik beschränkt ist sondern vielseitig angewendet werden kann. („Lebensvorbereitung“)
- Durch geeigneten Informatikunterricht erlangen die Schüler die Fähigkeit Gegenstände und Zusammenhänge schnell aufgrund ihrer Struktur zu erfassen und zu erlernen. („Lebensvorbereitung“, „Weltorientierung“ und „Stärkung des Schüler-Ichs“)
- Durch Bewertungskriterien und Problembetrachtungen gelangen die Schüler zu einer Mündigkeit im Umgang mit Informatiksystemen. („Anleitung zum kritischen Vernunftgebrauch“)
- Durch die kritische Betrachtung von Sicherheits- und Machbarkeitsaspekten wird die Grundlage für einen verantwortungsbewußten Umgang der Schüler mit Informatiksystemen gelegt. („Entfaltung von Verantwortungsbereitschaft“)
- Durch grundlegende Kenntnisse über Informatiksysteme gewinnen die Schüler an Sicherheit und lassen sich leichter auch auf den Umgang mit unbekanntem Systemen ein. („Stärkung des Schüler-Ichs“)

Wichtig dabei ist, dass der Begriff der Allgemeinbildung immer nur im Kontext der zugehörigen Gesellschaft gesehen werden kann.² Die hier geforderten Kompetenzen ergeben sich aus der Gesellschaft und sind somit veränderlich. Wird der Beitrag des Informatikunterrichts wie hier angegeben, so muss eine wissensorientierte Gesellschaft mit mathematischer, naturwissenschaftlicher und informatischer Ausprägung vorhanden sein.

3.Aufgabe:

Werden im Informatikunterricht Themen und Inhalte in die verschiedenen Bereiche der Fachwissenschaft aufgegliedert unterrichtet, so können folgende Schwierigkeiten entstehen:

<i>Schüler</i>	<i>Lehrer</i>
- Unmotiviert - Leicht abgelenkt - gelangweilt => Laut, stören, mit anderen Dingen beschäftigt	Unterrichtsinhalte werden vom Lehrer präsentiert => ähnliche Stundenabläufe, „theoretischer“ Unterricht
Lernt Inf. nicht als Einheit kennen, und gewinnt so erst spät einen Überblick.	Muss sich für Gewichtung der einzelnen Bereiche entscheiden und schafft so ein einseitiges Bild der Inf. .
- Gesamtbild der Situation existiert nicht - Kritischer Umgang mit Inf.Sys. erschwert - Anwendbarkeit und Nutzen unklar	Aufgaben werden genau auf einen Bereich zugeschnitten. => Realitätsfern, kein Praxisbezug

¹ „Informatik in der Sekundarstufe II, Vom Gesamtkonzept zum Curriculum – Planung von Kurssequenzen“ von Monika Seiffert aus „LOG IN Heft Nr.124 (2003)“

² <http://www.uni.landau.de/~infopaed/HoefflerMehlmer/Seminar%20Allgemeinbildung/Allgemein%201.rtf>.

<i>Schüler</i>	<i>Lehrer</i>
Vielschichtigkeit der Inf. bleibt verborgen.	Wie können Sinnvolle Schüleraufgaben gestellt werden, ohne mehrere Bereiche mit einzubeziehen?
Schüler muss den Zusammenhang der einzelnen Bereiche und Themen alleine finden. => Verständnis und Zugang zur Inf. erschwert.	Zusammenhang der einzelnen Bereiche wird nicht bis kaum Thematisiert.
Zusammenhang der einzelnen Inhalte wird nicht erkannt.	Motivation der Klassen schwindet durch mangelndes Verständnis.

Sinnvoller erscheint es von Beginn an ein Gesamtbild der Informatik zu erzeugen. Das bedeutet, dass die einzelnen Bereiche der Fachwissenschaft nicht isoliert im Unterricht betrachtet werden, sondern im Kontext der anderen. Um das zu erreichen sollte versucht werden Problemstellungen zu nutzen, welche die Schüler auf Fragen nach den anderen Bereichen lenken. Natürlich werden auch hier in einzelnen Unterrichtseinheiten nicht alle Bereiche thematisiert. Gerade bei der Einführung neuer Inhalte ist es notwendig in einem Bereich zu verbleiben. Der entscheidende Unterschied zu der anderen Struktur liegt darin, dass die Ausrichtung nicht ein unabhängiges Wissen in einzelnen Bereichen sondern ein zusammenhängendes Gesamtwissen ist.

Dabei gibt es zwei grundlegende Strukturen:

- Die Inhalte können in einem Problem erkannt und herausgearbeitet werden. Im Anschluss werden die so gewonnenen Erkenntnisse dann verallgemeinert und erweitert.
- Die Inhalte können isoliert vorgetragen und vertieft werden. Im Anschluss wird dann durch geeignete Aufgaben das so gewonnene Wissen angewendet und der Bezug der einzelnen Bereiche hergestellt.

4.Aufgabe:

Auffällig am vorgestelltem Gesamtcurriculum ist, dass die Anwendungsbereich in den Klassen 5 bis 10 konkreter gehalten sind, als in den Klassen 11 bis 13.

In der Sek. I wird zunächst auf die „bekanntem“ Anwendungsbereiche der Informatik eingegangen, auch wenn hier ein theoretischer Hintergrund geschaffen wird so steht doch der Umgang, die Schaffung und die Bewertung von Software im Vordergrund. Mit steigender Jahrgangsstufe kommen weitere Anwendungsbereiche und mit ihnen neue theoretische Grundlagen hinzu. Auch die Beurteilung von Informatiksystemen erhält eine andere Perspektive.

Insgesamt spiegelt sich im Gesamtcurriculum für die Sek. I der Umgang der Gesellschaft mit Informatiksystemen wieder. Die Schüler durchlaufen hier den Prozess den (fast) jeder private Nutzer von Informatiksystemen durchläuft. Sie beginnen bei den Anwendungsbereichen, die in unsere Gesellschaft als Zentral und Wichtig angesehen werden um darauf weiter aufzubauen ähnlich den Zusatzqualifikationen.

In der Sek. II lässt sich ein deutlich anderer Schwerpunkt ausmachen. Hier bilden die Anwendungsbereiche nur noch den Rahmen für die theoretische und kritische Auseinandersetzung mit Informatik. Den Schülern werden hier universelle Fähigkeiten vermittelt die nicht auf die Informatik beschränkt sind. Die Inhalte sind so ausgewählt, dass ein Verständnis der Strukturen und Wirkweisen ermöglicht wird.

5.Aufgabe:

Bei einer Lerngruppe die (vereinfacht) zwei verschiedene Wissensniveaus hat, besteht die Schwierigkeit darin die eine Gruppe nicht zu Unter- und die andere nicht zu Überfordern. Als Zielsetzung muss am Ende des Schuljahrs ein gleicher Wissenstand für die gesamte Lerngruppe stehen. Dabei ist darauf zu achten, dass keine der beiden Gruppen mehr Arbeit zu leisten hat als die andere.

Im Informatikunterricht kann mit dieser Schwierigkeit auf unterschiedliche Weise umgegangen werden. Ein Vorteil besteht darin, dass sich Lerninhalte und Themen wiederholen, auch wenn dabei auf verschiedenen Leistungsstufen gearbeitet wird. Hat man selber die Möglichkeit in das Geschehen der Sekundarstufe I einzugreifen, sollte man versuchen hier andere Programmiersprachen und Paradigmen als in der Sekundarstufe II zu nutzen.

Grundlegend sollte man versuchen durch gezielte Aufgabenstellungen und Gruppenaufteilungen dafür zu sorgen, dass die Schüler mit Vorkenntnissen diese den anderen zur Verfügung stellen und ihnen helfen. So kann bei der Einführung eines neuen Themas das Wissen der Klasse zu diesem Bereich abgefragt und festgehalten werden. Hierdurch haben alle Schüler eine gemeinsame Grundlage für die folgenden Stunden. Die Wissensabfrage fordert zunächst natürlich nur diejenigen die schon Informatik Unterricht hatten.

Durch die Räumlichkeiten sind meist noch Partnerarbeiten an vielen Stellen notwendig, dieses kann genutzt werden indem Partner mit verschiedenem Vorwissen kombiniert werden. Hierbei ist aber darauf zu achten das beide arbeiten können und nicht einer zusehen muss wie die Aufgabe zu lösen ist. Durch Regelungen kann z.B. festgelegt werden welche Aufgaben von welcher Schülergruppe (Vorkenntnisse Ja / Nein) erledigt werden. Hier haben die Schüler ohne Vorkenntnisse dann auch die Möglichkeit fehlende Erfahrungen im Umgang mit Computern gut zu machen.



Übung zur Vorlesung „Didaktik der Informatik II“

Blatt 2

Ein insbesondere in Österreich weit verbreiteter Ansatz zur informatischen Bildung besteht in der Zertifizierung durch den Europäischen Computer Führerschein™ (ECDL)³. In dieser Übung soll geprüft werden, inwieweit sich Aspekte dieses Konzeptes auf den Informatikunterricht in der Sekundarstufe II 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 Fragenkatalogs⁴ 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 Beispielprüfungen⁵ des ECDL. Welchem Anforderungsniveau entsprechen die Prüfungsfragen? Können Sie sich vorstellen, dass Schülerinnen⁶ ihrer zukünftigen Schule zum Ende der Jahrgangsstufe 11 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.

³ <http://www.ecdl.at>

⁴ <http://www.gym1.at/schulinformatik/aus-fortbildung/fachdidaktik/kv-01/grundlagenfragen.pdf>

⁵ http://www.ecdl.at/syllabus/beispiel/Beispiel-Pruefungen/Beispiel_Pruefung.pdf

⁶ Auch auf diesen Übungsblättern wird das generische Femininum verwendet. Die männliche Schreibweise ist implizit mit gemeint.

Aufgabe 1:

a) Historie

1994 wurde durch die finnische Computergesellschaft (FIPA - Finnish Information Processing Association) eine Erhebung durchgeführt, die die Bedürfnisse der Wirtschaft an informatischer Bildung aufzeigen sollte. Daraus wurde eine siebenteilige Modulstruktur entwickelt, die später zu der des ECDLs werden sollte.

Dieses wurde dann 1995 durch den internationalen Dachverband der Europäischen Informatikgesellschaften (CEPIS - Council of European Professional Informatics Societies) aufgegriffen und durch eine Arbeitsgruppe weiterentwickelt.

1996 wurde diese Arbeitsgruppe in die ECDL Foundation umgewandelt, die im selben Jahr ein Pilotprojekt mit dem ECDL in Schweden durchführte.

1997 wurde der ECDL daraufhin in vielen europäischen Ländern durch nationale Organisationen eingeführt.

b) Lernziele und Prüfungsinhalte

Allgemein bestehen die Lernziele des ECDLs in anwendungsbezogenen Fertigkeiten. Es soll der Umgang mit dem Computer erlernt werden. Die Praxisorientierung ergibt sich (wie schon unter a) erwähnt) aus den Bedürfnissen der Wirtschaft an informatischen Fähigkeiten. Der ECDL ist in sieben Module unterteilt, auf die ich nicht einzeln eingehen möchte, da sich die Inhalte der Module in drei wesentliche Bereiche zusammenfassen lassen (die im übrigen nicht an den Modulen orientiert sind):

Erläuterung von Begriffen

Hiermit sind grundlegende Begriffe für den Umgang mit Computern und deren Software gemeint. Es soll gelernt werden, was diese Begriffe anwendungsbezogen bedeuten. Ihre Erläuterung bezieht sich also auf ihre Wechselwirkungen und Funktionen im Anwendungskontext. Also z.B.: Was sind die einzelnen Komponenten eines Computers? Wozu braucht man sie? Was beeinflussen sie?

Anwendung einzelner Programme

Es soll gelernt werden, wie mit einzelnen Anwendungsprogrammen (das sind: Büro-Anwendungen, Internet-Anwendungen und die Desktopumgebung) umgegangen werden soll. Das heißt, wie man bestimmte Anwendungsziele - etwa einen Serienbrief erstellen - mit einer Software erreicht.

Informatik und Gesellschaft

Es sollen grundlegende Aspekte, die für die Wirtschaft von Belang sind, der Informationssicherheit, Gesundheit, Gesellschaft und des Rechts gelernt werden. Dabei geht es vorwiegend um die Wichtigkeit bzw den Nutzen von diesen. Z.B.: Wozu sind Zugriffsrechte wichtig?

c) Zusammenfassung und Bewertung (Was fehlt?)

Man merkt dem ECDL deutlich an, dass er auf die Bedürfnisse der Wirtschaft zugeschnitten ist. Diese erwarten von ihren Mitarbeitern, dass sie mit einem Computer verantwortungsvoll umgehen können. Dabei geht es lediglich um die Benutzung einzelner Programme und die von Netzwerken. Die Grundlagen beschränken sich so auch auf die für die Bedienung eines Computers wichtigen. So wurden sie von mir in den Lernzielen auch nur als Begriffserklärung aufgenommen, da sie sich ansonsten nur auf Wechselwirkungen zueinander und zu Anwendungsprogrammen beziehen. Aspekte aus dem Bereich Informatik und Gesellschaft werden nur in soweit aufgenommen, wie es für Firmen und Behörden wichtig ist, das ist im wesentlichen Datensicherheit, Urheberrecht sowie Gesundheitsaspekte, soweit letztere für Produktivitätssteigerung wichtig sind.

Im Bezug auf den Informatikunterricht in der Sekundarstufe II vermissen vor allem die korrekte Vermittlung von Grundlagenwissen. Bei den Begriffserläuterungen leistet sich der ECDL ein Quidproquo zwischen dem Begriff einer Sache und seiner Funktion¹. (Auch wenn dies bei Computern nur sehr schwer zu trennen ist.)

¹ Dies habe ich aus der Beispiel Prüfung abgeleitet. In dem Anforderungspapier erkennt man dies nicht auf Anhieb.

Da die Sek. II auf wissenschaftliches Arbeiten vorbereiten soll, kann man diese so nicht in den Unterricht übertragen.

Es fehlen Grundlagen der Computertechnik und -architektur, diese müssen zwar nicht im Detail behandelt werden, ich halte es jedoch ein grundlegendes Verständnis für wichtig, um Gründe für die Wechselwirkungen von Komponenten eines informatischen Systems zu erkennen.

Die konzeptionellen Hintergründe von informatischen Systemen (Software und Hardware) fehlen vollständig. Dadurch fördert man kein Verständnis, vielmehr fordert es gerade dazu auf, die Inhalt einfach auswendig zu lernen. Das kann nicht ein Ziel von Unterricht sein, vielmehr will man ja ein Wissensstand erreichen der gerade unabhängig vom konkreten informatischen System ist, zur Erschließung andere herangezogen werden kann und evtl. in außer informatische Bereiche übertragen werden kann. Dazu müssen Konzepte, wie z.B. die objektorientierte Modellierung (u.v.m.), vermittelt werden.

Außerdem fehlen die Zusammenhänge zwischen einzelnen informatischen Bereichen.

Nicht zuletzt fehlen wichtige Aspekte aus dem Bereich Informatik und Gesellschaft, wie Wechselwirkung zu Individuen und der Gesellschaft, Auswirkungen von Computernetzen uvm.

Aufgabe 2²:

1. Frage: Was ist ein Datensatz?

- (a) Eine Adresse
- (b) Eine logisch zueinander gehörende Menge von Daten
- (c) Mehrere unterschiedene Daten, die zueinander in einer Verbindung stehen und die jeweils einer Objektart zu geordnet werden können.
- (d) Ein Datum, das einer Objektart zu geordnet werden kann.

2. Frage: Was versteht man unter einer Festplatte für einen PC?

- (a) Einen nicht flüchtigen Datenspeicher
- (b) Eine Komponente eines Computers, ohne die er nicht funktionieren würde.
- (c) Ein Medium, das fest im Computer eingebaut ist.
- (d) Ein optischer Datenträger

3. Frage: Nennen Sie Situationen, in denen ein Mensch besser ist/arbeitet als ein Computer!

- (a) Wenn komplexe mathematische Berechnung erstellt werden sollen.
- (b) Wenn ein Text in eine andere Sprache übersetzt werden soll.
- (c) Beim Erkennen von Sprache
- (d) Wenn Maschinen gesteuert werden sollen.

4. Frage: Was ist ein lokales Netzwerk!

- (a) Ein Zusammenschluss mehrerer Computer auf den nur eine bestimmte Personengruppe Zugang hat.
- (b) Das Internet
- (c) Ein örtlich begrenzter Zusammenschluss von Computern.
- (d) Eine Organisationsstruktur des Internets.

Ein Vorteil von Multiple-Choice-Aufgaben liegen im geringen Korrekturaufwand, wobei jedoch durch das Erstellen von sinnvollen Antwortmöglichkeiten, die den gewünschten Schwierigkeitsgrad haben, ein Teil dieses Vorteils verloren geht. Die Bewertungskriterien sind für die Schüler klarer erkennbar. Es ist klarer erkennbar, ob ein Schüler das Abgefragte verstanden hat. Allerdings lassen sich komplexere Aufgabe kaum durch das Multiple-Choice-Verfahren erstellen. Es lässt sich mit ihnen kaum feststellen, ob ein Schüler die Zusammenhänge des Abgefragten verstanden hat. Daraus ergibt sich ein Einsatzschwerpunkt des Multiple-Choice-Verfahrens in Fragen die auf Begriffserläuterung zielen. Es lässt sich mit ihnen auch ein einfaches Faktenwissen abfragen. Auch ein methodisches Vorgehen kann abgefragt werden.

Bei den Freitext-Aufgaben verhält es sich nahezu umgekehrt, mit ihnen lassen sich gut komplexer Zusammenhänge abfragen. Die Bewertung von Aufgabe wird jedoch schwieriger und für Schüler undurchsichtiger. Es wird beim Abfragen von reinen Faktenwissen schwieriger zu entscheiden, ob eine Aufgabe richtig oder falsche beantwortet wurde.

²Die aus dem angegebenen Fragenkatalog ausgewählten Fragen können hier evtl. leicht verändert sein, um daraus ein Multiple-Choice-Aufgabe zu erstellen. Ich habe dabei darauf geachtet, den Inhalt des Gefragten nicht zu ändern.

Aufgabe 3³:

Wie schon in Aufgabe 1 erwähnt, entsprechen die Anforderungen des ECDLs, im wesentlichen die der Benutzung einzelner Programme. Es soll nachgewiesen werden, dass jemand fähig ist, einzelne Programme zu benutzen. Dabei geht es in den Prüfungsfragen um das Abfragen von Anwendungsbeispielen; dass jemand ganz bestimmte Sachen mit einem Computer machen kann. Sie entsprechen also (wie schon gesagt) eher dem Auswendiglernen von Anwendungsabläufen.

Ich glaube nicht, dass meine Schüler⁴ diese Prüfung bestehen und auch sollten sie dies nicht. Beim ECDL geht es primär um das Abfragen von Anwendungsabläufen, die eine Menge Übung an den entsprechenden Anwendungen erfordert. Da es mir aber im Unterricht um das Verständnis von informatischen System (evtl. auch von Anwendungsprogrammen) geht, wäre dies kein adäquates Mittel, da das Verständnis nicht durch das (Auswendig)lernen von Anwendungsabläufen bzw. das Üben an konkreten Anwendungen gefördert wird, eher im Gegenteil - es verschleiert die strukturellen Hintergründe. Es sollen informatische Konzepte, Modelle und Modellierungstechniken im Unterricht gelernt werden. Das kann man selbstverständlich an einzelnen Anwendungsbeispielen thematisieren; evtl auch an bestehender Anwendungssoftware. Der Grund für diese Zielsetzung liegt auf der Hand: Anwendungen ändern sich, es gibt zigtausende Anwendungen, viele verschiedene informatische Systeme. Mit dem Wissen kann ein Schüler dann mehr als nur eine Anwendung bedienen, er kann sie sich, sofern sie korrekt entwickelt wurden, erschließen. Dieses Erschließen allein wird nur nicht dazu reichen die ECDL-Prüfungen zu bestehen.

Kurz: Meine Zielsetzung für den Unterricht ist grundlegendes Verständnis möglichst vieler informatischer Systeme (sowohl Hardware als auch Software). Das des ECDLs ist es einzelne bedienen zu können. Und diese beiden Zielsetzung sind diametral zueinander.

Daraus ergibt sich, dass die Prüfung für den Unterricht, bis auf ein paar Begriffsfragen nicht zu gebrauchen ist. Welche Bereiche ich hinzufügen würde habe ich bereits oben in dieser Aufgabe und in Aufgabe 1 erwähnt. Um konkreter zu werden eine paar Schlagwörter:

- Datenanalyse
- Modellierung und Gestaltung von Benutzeroberflächen
- Objektorientierte Modellierung
- Zusammenwirken (bzw, Wirkungen auf) von informatischen System und der Gesellschaft, sowie des Individuums
- Architekturen von Computern
- Strukturen in/von Computernetzwerken

Dies um nur einige konkrete Bereiche zu nennen. Außerdem fehlt es der ECDL-Prüfung an Vermittlung zwischen den einzelnen Bereichen, eine Erklärung der Wechselwirkung reicht dazu nicht aus - sie stehen fast zusammenhanglos neben einander. Zu den Gründen, die ich oben bereits genannt habe, kommt noch das die Konzepte der Informatik übertragbar in andere Bereiche (ausserinformatisch) sein soll. Keines der genannten Ziele lässt sich durch die Prüfungsfragen des ECDLs überprüfen.

³Teile aus Aufgabe 1 c) treffen auch hier zu bzw. sind dort schon genauer erläutert und teilweise begründet.

⁴Anm.: Hiermit sind selbstverständlich auch weibliche Schüler gemeint. Schüler ist ein Oberbegriff weswegen man das auch eigentlich nicht immer dazu sagen muss. Ich glaube nicht das man dem Bestreben von gesellschaftlichen Gruppen nach Einfluss in den Sprachgebrauch nachgeben muss. Aber das ist ein anders Thema...:)

Übung DDI II Blatt 2

1. Aufgabe:

Hinter der Abkürzung ECDL verbirgt sich ein Prüfungssystem mit anschließendem Zertifikat. Die Idee ist es eine Aussage über die Fähigkeiten und das Wissen des Einzelnen im Umgang mit Computern treffen zu können.

1994 in der Entstehungsphase dieses Prüfungssystems standen die „Bedürfnisse der Wirtschaft“⁴ im Vordergrund. Erst später wurde dieses Projekt von einer selbständigen Organisation weiterentwickelt und bis 2003 in mehreren Europäischen Ländern vorgestellt.

Der ECDL ist in 7 Stufen aufgliedert die eigene Lernziele und Prüfungsinhalte enthalten. Dabei soll das Wissen der einzelnen Stufen aufbauend sein.

<i>Modul</i>	<i>Lernziel</i>	<i>Prüfungsinhalt</i>	<i>Fehlende Zieldimension</i>
1	Grundlagen der Informationstechnologie in den Bereichen: - Hardware - Software - Informationsnetze - Informationstechnologie im Alltag - Gesundheit, Sicherheit und Umwelt - Urheberrecht und Gesetzeslage	- Wissensabfrage durch Multiple-Choice-Test aus den Bereich der Lernziele. - Grundlagenfragen die kein Themenverständnis voraussetzen. - Anwendungsorientierte Fragen	- Aus den Prüfungsfragen kann keine Aussage über das Verständnis des Einzelnen zu den Themen getroffen werden - Anwendungs- aber nicht Verständnisorientiert - Verständnis und Hintergrund zu den Themen fehlt
2	Computerbenutzung und Dateimanagement in den Bereichen: - Arbeitsumgebung am Computer - Schreibtisch/Desktop - Dateien verwalten - Viren - Druckverwaltung		
3 / 4 / 5 / 6 / 7	Umgang mit einer Software zur Textverarbeitung/ Tabellenkalkulation/ Datenbanken / Präsentationen / Internet oder E-Mail: - Programm benutzen - Grundlegende Arbeitsschritte - Formatierung - Formeln und Funktionen - Information abfragen - Effekte - Diagramme - Suchen - Ausgabe vorbereiten	- Anwendung der Softwareprodukte um vorgegebene Aufgaben zu erledigen - Abfrage von Handlungsabläufen - Abfrage von Befehlen	- Zusammenhänge und Struktur der Software werden nicht vermittelt - Obwohl zwischen den Stufen 3 bis 7 Parallelen bestehen, die es möglich machen sie als eine Stufe zu betrachten, wird hier ein Aufdecken der Gemeinsamkeiten vermieden. - Der Lernende wird nicht zur Selbständigkeit und Mündigkeit erzogen. - Kenntnisse, die den Lernenden dazu befähigen, selbständig zu lernen werden vermieden.

¹ [Http://www.ecdl.at](http://www.ecdl.at)

2.Aufgabe:

	<i>Vorteile</i>	<i>Nachteile</i>	<i>Einsatzschwerpunkte</i>
Freitextaufgabe	<ul style="list-style-type: none"> - Lernende können nur Antworten erstellen, wenn sie etwas über das Thema wissen - das gesamte Wissen kann präsentiert werden - Antworten werden durch Begründungen nachvollziehbar - Bessere Möglichkeit, Defizite zu erkennen und aufzuarbeiten - das tatsächliche Wissen wird dargestellt 	<ul style="list-style-type: none"> - hoher Korrekturaufwand - das Thema wird leicht mit der Antwort verfehlt - hoher Zeitaufwand für die Bearbeitung - Bearbeitungszeit der Einzelnen ist sehr unterschiedlich - Vergleich der einzelnen Arbeiten ist schwierig 	<ul style="list-style-type: none"> - Gesamte Wissensabfrage zu einem Thema abfragen - Verständnisfragen
Multiple-Choice-Aufgabe	<ul style="list-style-type: none"> - leichte Korrektur - kurze Bearbeitungszeit - fast identische Bearbeitungszeiten bei allen Lernenden - Vergleichbarkeit der einzelnen Arbeiten ist gewährleistet 	<ul style="list-style-type: none"> - ohne Wissen über das Thema kann die Prüfung durch raten bestanden werden - Formulierungen der Fragen und Lösungen sind darauf ausgelegt falsch verstanden zu werden - Wissenstand ist damit NICHT abfragbar - Es kann nicht nachvollzogen werden wie es zu falschen Antworten kam - Defizite sind nur schwer zu erkennen 	<ul style="list-style-type: none"> - Auswahlverfahren: Um Einzelne aus einer Gruppe auszuwählen - Große Gruppe: Um den Korrekturaufwand und die Vergleichbarkeit bei der Überprüfung großer Gruppen zu gewährleisten. - Beispiele: Führerscheinprüfung Wissensabfrage an Schulen und Universitäten Eignungstest

Ich halte es nicht für sinnvoll, die Grundlagenfragen² in Form von Multiple-Choice- Aufgaben zu stellen. In den Beispielaufgaben der ECDL sieht man ganz deutlich, dass durch die Auswahl der Antworten ein Großteil der Multiple-Choice-Fragen durch Ausschließen richtig beantwortet werden kann.

I) Welchen Zweck hat das Datenschutzgesetz?

- a) Schutz der Bürger über die personenbezogene Daten mit Informationstechnik verarbeitet werden.
- b) Schutz des Informatiksystems (Computer).
- c) Schutz der Datensätze in einem Informatiksystem.
- d) Schutz der Verantwortlichen die mit Informatiksystemen arbeiten.

II) Welche der Aussagen trifft auf eine Datenbank zu?

- a) Wird nur für die Erfassung von Datenmengen genutzt.
- b) Wird genutzt zur Beschreibung, Speicherung und Wiedergewinnung von Datenmengen.
- c) Gehört immer zu mindestens einem Anwendungsprogramm.
- d) Schließt aus, dass verschiedene Programme den gleichen Datenbestand nutzen können.

III) Was versteht man unter einem vernetztem Informatiksystem?

- a) Die Verkabelung der einzelnen Geräte eines Computers miteinander (Festplatte, Laufwerke, Bildschirm, Tastatur, Maus, Drucker, ...).

² [Http://www.gym1.at/schulinformatik/aus-fortbildung/fachdidaktik/kv-017grundlagenfragen.pdf](http://www.gym1.at/schulinformatik/aus-fortbildung/fachdidaktik/kv-017grundlagenfragen.pdf)

- b) Den Zusammenschluß mehrerer Informatiksysteme, so dass auf die Geräte und Daten der anderen zugegriffen werden kann.
 - c) Ausschließlich das Internet.
 - d) Die netzartigen Verbindungen auf den inneren Bausteinen von Informatiksystemen.
- IV) Was versteht man unter *Einfügen* in „Büro orientierter Software“?
- a) Eine Datei wird in einen vorhandenen Ordner eingefügt.
 - b) Objekte werden in ein vorhandenes Dokument eingefügt.
 - c) Ein markierter Bereich wird verändert.
 - d) Objekte werden in einen markierten Bereich eingefügt und die vorher markierte Bereich wird gelöscht.

3.Aufgabe:

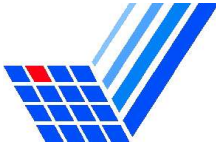
Sollten die Anforderungen der Beispielprüfungen, mit denen der tatsächlichen des ECDL übereinstimmen, so kann ich nicht nachvollziehen, wie hier von einem standardisierten und grundlegendem Bildungsniveau für Computeranwender gesprochen werden kann. Wenn nach einem grundlegendem Bildungsniveau gefragt wird, müssen dort Kenntnisse und Fähigkeiten erworben werden, die das Verständnis und den Umgang von und mit Informatiksystemen fördern.

Die hier abgefragten Kenntnisse können ohne Mühe durch „das Spielen“ mit dem Computer und der entsprechenden Software erlangt werden. Aber nutzen sie etwas für den späteren Umgang mit Informatiksystemen?

Es scheint wenig hilfreich, wenn gezielte Handhabungen trainiert werden, die abhängig von den genutzten Produkten variieren. Unter Grundlagen versteht man gemein hin Dinge, die helfen, die darüber liegenden Bereiche zu erreichen und zu verstehen. In der Informatik der Sek I müssen Grundlagen also Kenntnisse und Fähigkeiten sein, die helfen die Stoffgebiete der Sek II zu erarbeiten und damit einen Weg zu einer „informatischen Mündigkeit“ weisen.

Anders aufbereitet und abgefragt enthalten die Modul 1, 2, 5 und 7 Inhalte, die eine gute und sinnvolle Grundlage sind. Würden die Inhalte der Module neu angeordnet, so dass z.B. der Aspekt Datenschutz zusammen mit Datenbanken auftaucht könnte hieraus ein gutes Konzept für die Sek I entstehen. Bei den Prüfungsfragen müssten Anwendungsorientierte Fragen Verständnisfragen weichen und auch die Antworten im Multiple-Choice- Bereich müssten überarbeitet werden.

Insgesamt müsste ein Verständnis von Zusammenhängen angestrebt werden was durch geeignete Tests abgefragt würde.



Übung zur Vorlesung „Didaktik der Informatik II“

Blatt 3

In dem Beitrag „Methodenkompetenz im Informatikunterricht“ von Lutz Engelmann¹ werden vom Autoren Ansätze aufgezeigt, Anwendungsprogramme mit Hilfe der Objekt-Attribut-Methode im Informatikunterricht zu nutzen. Lesen Sie für die Bearbeitung dieses Übungsblattes den Beitrag.

1. Aufgabe: (3 Punkte)

Zeigen Sie auf, wie sich die Verknüpfung der Objekt-Attribut-Methode mit Anwendungsprogrammen im Informatikunterricht der Sekundarstufe II begründen lässt und skizzieren Sie die Verknüpfung für mindestens drei Arten von Anwendungsprogrammen.

2. Aufgabe: (2 Punkte)

Welche Vorteile und welche Nachteile hat eine objektorientierte Betrachtungsweise bei der Benutzung von Anwendungsprogrammen? Betrachten Sie darüber hinaus auch, welcher Nutzen in Bezug auf andere Informatikinhalte entstehen kann.

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 Lutz Engelmann: Methodenkompetenz im Informatikunterricht, LOG IN 121 (2003).

Didaktik der Informatik II -- Lösung zu Blatt 3

Aufgabe 1:

a) Begründung der Methode

Es gibt eine Reihe von Gründen, die für die Verknüpfung der Objekt-Attribut-Methode zu Anwendungsprogrammen sprechen:

1. Anwendungsprogramme sind meist objektorientiert aufgebaut. Ihre Benutzeroberfläche ist nach Objekt-Attribut Gesichtspunkten aufgebaut. Die Eigenschaften eines Objektes der Anwendung werden im entsprechenden Objektmenü (z.B. Kontext-Menüs) eingestellt bzw. angezeigt. Vermutlich sind die meisten Anwendungsprogramme auch objektorientiert programmiert worden und diese Methode entspricht einem Teil der objektorientierten Modellierung. Also entspricht diese Erklärungs- methode im wesentlichen der internen und nach außen sichtbaren Struktur von Anwendungsprogrammen -- die Methode erklärt also die Struktur von Anwendung und nicht einfach nur die Funktionen; die Anwendung wird aus sich selbst heraus erklärt.
2. Ist eine Art von Anwendungen der Struktur nach erklärt, wie es durch diese Methode gewährleistet ist, lässt sich diese auf andere Anwendungen gleicher Art übertragen. Die Schüler können so ein vom konkreten Anwendungsprogramm unabhängiges Wissen erlangen.
3. Wird auch diese Methode, wie es sich anbietet, mitvermittelt, können sie Schüler auf nahezu jedes Anwendungsprogramm anwenden. Damit sind sie in der Lage, sich auch unbekannte Anwendungen und Anwendungsarten zu erschließen. Da es sehr viel verschiedene Anwendungen und Arten von ihnen gibt, ist das ein nicht zu unterschätzender Vorteil. Außerdem wird wahrscheinlich jeder Schüler irgendwann, abhängig von seiner Berufswahl, mit Spezial- bzw Individualsoftware konfrontiert werden, die er sich dann auch mit dieser Methode erklären kann.
4. Da diese Methode schon Teile der objektorientierten Modellierung enthält, kann für die Erklärung der selben im Informatikunterricht darauf aufgebaut werden. So kann man die Modellierung auch stufenweise einführen: zunächst nur Objekt(Klassen)-Attribut Beziehungen, darauf folgen zusätzlich Methoden und schließlich Beziehungen von Klassen (Vererbungen, Assoziationen, Aggregationen, Kompositionen). Und dies kann an konkreten Anwendungen erfolgen, was den Vorteil hat, dass Schüler unmittelbar sehen können wofür die Modellierung gebraucht werden kann. Von da aus kann dann der Übergang zu selbst Entwicklung kleiner Programme/Projekte erfolgen. Außerdem ist es leichter, die Objektorientiertheit zu verstehen, wenn sie an einem Anwendungsprogramm, das schon vorliegt, erklärt wird.
5. Auch wenn dies meisten Objekte der Informatik außerhalb von ihr nicht existieren, kann ein Übergang in die "Realewelt" erfolgen. Die Methode eignet sich nämlich auch zum erklären der Welt. Die Bestimmung einer Sache ist logisch gestehen nichts anders als ihre notwendigen Eigenschaften zu erkennen. Aus diesen Bestimmungen kann man dann durch logisches Schließen Erscheinung erklären. Die Eigenschaften einer Sache sind in der Analogie nichts anders als Attribute.
6. Es ist sicher wichtig, dass Schüler mit Anwendungsprogrammen umgehen können. Sie werden dieses im späteren Leben und auch teilweise schon in der Schule brauchen.
7. Die Anwendungsprogramme werden informatisch erklärt. Die Methode erklärt die Struktur der Programme. Diese entspricht einer informatischen Analyse, ein Programm wird als informatisches System behandelt und durch seine Objektstruktur erklärt.

b) Die Verknüpfung für verschiedene Arten von Anwendungsprogrammen

Die jeweiligen Objekte habe ich hier einzeln aufgelistet. Natürlich kann man das in fast beliebige Tiefen fortsetzen, deswegen erhebe ich kein Anspruch auf Vollständigkeit; es sind nur jeweils die wichtigsten aufgeführt.

i) Mail-Programme

Die Objekte und Attribute eines Mail-Programm sind hier jeweils Aufgelistet:

- * Nachrichten:
- * Empfänger (kann weiter unterteilt werden in Rechner und Benutzer)
- * einen Absender
- * einen Betreff
- * Anhänge
- * einen Nachrichtentext
- * ein Flag für gelesen und ungelesen
- * Nachrichtenordner:
- * Ordnername
- * Anzahl enthaltener Nachrichten gelesen und ungelesen
- * Beschreibung zum Ordner
- * Adressen:
- * Name
- * Vorname
- * E-Mail-Adresse
- * Spitzname
- * evtl. Beschreibung
- * Adressbücher:
- * Adressbuchname
- * Anzahl enthaltener Adressen
- * Beschreibung

ii) DTP-Programme

Hier gibt es folgende Objekte mit den jeweiligen Attributen:

- * Dokument:
- * Name
- * Erstellungsdatum
- * Dokumentart (Buch, Artikel usw.)
- * Format (Seitengröße, Rand usw.)
- * Standart-Seite (Vorlagen-Seite)
- * Format
- * Kopf- und Fuß
- * Spaltenanzahl
- * linke-, rechte-Seite
- * Seite
- * Zahl
- * linke, rechte-Seite
- * Spaltenanzahl
- * Text
- * Ausrichtung, Position
- * Überschrift, mit Schriftgröße

- * Schriftart
- * Schriftgröße
- * Grafiken
- * Ausrichtung, Position
- * Unter/Überschrift
- * Größe
- * Textumlauf

iii) 3D-Modellierungs-Programme

- * Flächen
- * Position der Ankerpunkte
- * Flächenfarben
- * Texturen
- * Transformationen (Position des gesamten Objektes)
- * Vielecke
- * Position der Eckpunkte
- * Flächenfarbe
- * Texturen
- * Transformationen (Position des gesamten Objektes)
- * Kugeln bzw Ellipsoide
- * Radien
- * Flächenfarbe
- * Texturen
- * Transformationen (Position des gesamten Objektes)
- * Schrift
- * Schriftgröße
- * Schriftart
- * Tiefenausdehnung
- * Flächenfarbe
- * Texturen
- * Transformationen (Position des gesamten Objektes)

Aufgabe 2:

Siehe hierzu auch Aufgabe 1. Die Gründe sind auch zugleich Vorteile. Auch der Nutzen für Informatikinhalt ist dort schon erwähnt.

a) Vor- und Nachteile für die Benutzung von Anwendungsprogrammen

Für die Schüler sollte es möglich sein durch diese Methode, sich die meisten Anwendungsprogramme zu erschließen, da die Programme meist entsprechend aufgebaut sind. Die Unabhängigkeit von der konkreten Anwendung ist auch ein entscheidender Vorteil, denn das Wissen über eine Anwendungsart ist übertragbar auf andere Anwendungsprogramme unterschiedlichster Hersteller und Plattformen. Hat man sich einmal mit einer Art von Programmen methodisch auseinandergesetzt, dürfte es ein Leichtes sein, dieses gewonnene Wissen auf ein anderes konkretes Programm zu übertragen. Die Einarbeitungszeit in Programme wird so stark verkürzt. Je mehr Anwendungsprogramme so analysiert wurden, um so leichter wird das Erlernen eines neuen Programms bzw einer neuen Art, denn erstens wird die Methode geübt und zweitens sind viele Konzepte in verschiedenen Programmen gleich.

Allerdings kann es vorkommen, dass Programme nicht nach den Regeln der Objekt-Attribut-Methode aufgebaut sind. Ein Zugang zu solchen Anwendungen mit dieser Methode ist zum Scheitern verurteilt. Wobei sich dieses an der in den Übungen viel diskutierten Kritikfähigkeit relativiert. Ein Programm

sollte vernünftig strukturiert aufgebaut sein.

Die Methode ist sicher nicht ganz leicht zu benutzen, da sie erstens schwierig zu erlernen ist. Sie benötigt einige Übung. Zweitens können Objekt- und Attribut-Zuordnungen auf verschiedenen Arten und Weisen erfolgen, wobei es dabei teilweise auf die Sichtweise ankommt. Diese unterschiedlichen Zuordnungen sind durchaus nicht als falsch zu bewerten. Es sei daran erinnert, dass wenn verschiedene Informatiker ein und das selbe Programmsystem unabhängig von einander entwickeln sollen, unterschiedlich modellierte Programme dabei heraus kommen. Diese sind zwar ähnliche aber nicht gleich. Allerdings kann dies bei der Benutzung des Programms zu Problemen führen, wenn verschiedene Zuordnungen getroffen worden sind.

Ein entscheidender Vorteil ist, dass Programme korrekt benutzt werden können, d.h. so wie es eigentlich vorgesehen war. Dies führt zu einem besseren Umgang, und auch schneller zum gewünschten Ergebnis. Ist z.B. das Konzept von Absätzen in Textverarbeitungen nicht bekannt, kann daraus eine nicht gewollte Formatierung des Textes folgen. Außerdem kann jemand durch ein Wissen über die Struktur eines Programmes viele Zusatzfunktionen benutzen, die er ohne dieses nicht Verstehen würde und vermutlich auch nicht oder falsch benutzen würde. Diese können einem -- wie etwa Dokumentenklassen -- erheblich die Arbeit erleichtern (und dazu sollten Computer doch eigentlich da sein). Z.B. will jemand der einen Text schreibt, sich nicht oder nur wenig um die Formatierung kümmern -- er will halt nur einen Text schreiben, diese kann einem mit dem Objektgedanken weitestgehend ein Computer abnehmen, indem man dem Programm "mitteilt" was für eine Struktur der Text haben soll.

b) Nutzen für Informatikinhalt

An den Anwendungsprogrammen kann (wie oben schon erwähnt) das Konzept der objektorientierten Modellierung erklären. Die Objekt-Attribut-Methode ist ja der objektorientierten Modellierung entlehnt. Von der Methode ausgehend kann man also zum Gesamtkonzept der Modellierung gelangen.

Da die Modellierung von tatsächlichen Softwareprojekten schon ein weitgehendes Verständnis der Modellierungs-Strukturen und -Möglichkeiten voraussetzt, ist es von großem Nutzen dies bereits an Anwendungsprogrammen "geübt" zu haben. Damit, sollte man im Unterricht ein Softwareprojekt vor haben, dieses nicht an mangelnder Übung scheitert, was ziemlich fatal wäre, da die Schüler ziemlich enttäuscht wären, und den Sinn dieser Methoden nicht erkennen würden. Ein Fehlschlag, also eine nicht hinreichende oder falsche Objekt-Attribut-Zuordnung, ist auf einen Anwendungsfall beschränkt. Es sind außerdem Modellfehler wesentlich schneller, beim Anwenden der Software zu erkennen. Sollen die Schüler selbst eine Software erstellen, würde dies erst nach Fertigstellung erkennbar sein.

Die Sinnhaftigkeit von Modellierungsmethoden wird schon an der Anwendungssoftware sichtbar.

Die vorgestellte Methode ist auch für Datenanalyse bzw -modellierung relevant. Denn dies wird ja schon fast an den Anwendungen vollzogen. Auch die Analyse von Graphischen Oberflächen ist dabei schon in begriffen. Wobei jeweils der Nutzen dieser Methoden schon vor ihrer rein methodischen Befassung klar sein sollte, da die Schüler diesen schon am Anwendungsfall feststellen können.

Außerdem fördert dies Methode das Verständnis von Schnittstellen, die zum Austauschen von Dokumenten und zum Nachrichtenaustausch zwischen verschiedenen Anwendungsprogrammen benutzt werden können. Man kann sie auch erstmal, ohne ein Programm entwickeln zu müssen, benutzen. (Ich denke dabei an das Desktop Communication Protocol -- DCOP -- von KDE3).

Aufgabe 3:

Bezüglich des ECDL-Konzeptes siehe auch in meiner Abgabe zu Aufgabenblatt 2

Beide Konzepte haben zum Ziel, das Schüler mit Standardsoftware umgehen können. Die Bedienung des Computer und der Umgang mit ihm soll bei beiden erlernt werden.

Der Ausgangspunkt ist bei beiden auch der selbe: Von den Schülern wird später erwartet das sie Anwendungsprogrammen benutzen können. Wobei Lutz Engelmann dabei zusätzlich noch die Schüler selber sieht, von den Abiturienten meinen 41% ihre Kenntnisse im Umgang mit Computern seinen nicht ausreichend.

Wie die Konzepte dieses Ziel erreichen wollen darin unterscheiden sie sich erheblich:

Der ECDL

Die Vermittlung erfolgt an einem konkreten Anwendungsprogramm. Es werden jeweils einzelne Anwendungsfälle durchgespielt und abfragt. Dabei geht es in der Hauptsache, Funktionen einer einzelnen Software zu vermitteln. Diese Funktionen werden dann geübt und in den Prüfungen abgefragt. Die Orientierung der Inhalte ist eindeutig an den Bedürfnissen der Wirtschaft orientiert (auch wenn man auf den Internet-Seiten des ECDLs immer etwas von Unabhängigkeit lesen muss).

Die Objekt-Attribut-Methode siehe dazu auch oben

Im Gegensatz zum ECDL wird hierbei versucht, die Struktur der Anwendungsprogramme zu erklären. Weitergehender soll eine Methode zur Erklärung von Software allgemein vermittelt werden. Diese Erklärung soll dazu dienen, mit Software umgehen zu können. Außerdem soll sie einen vernünftigen und korrekten Umgang mit Programmen fördern. Dieses soll durch ein Verständnis der Struktur von Software erreicht werden.

Zusätzlich zum reinen Umgang mit Programmen, sollen mit ihr auch ein Schluss auf andere Informatikinhalt und Bereiche -- wie informatische Modellierung von Software und Benutzerschnittstellen sowie Schnittstellen von Programmen untereinander -- möglich sein.

Desweiteren soll mit diesem Konzept wissenschaftspropädeutisch im Informatikunterricht gearbeitet werden können.

Dieses soll durch die Erklärung von anwendungsspezifischen Objekten mit den zugehörigen Attributen erreicht werden. Damit soll die Struktur der Daten, wie sie die Anwendung verarbeitet, aufgezeigt werden. Auch Schnittstellen von Programmen können so analysiert werden. Da sich die Methode stark nach der objektorientierten Programmierung richtet, ist der Übergang zu anderen Inhalten der Informatik gewährleistet.

Aufgabe 4:

Bezüglich des ECDL-Konzeptes siehe auch in meiner Abgabe zu Aufgabenblatt 2

Um es hart und auf den Punkt auszudrücken, könnte man sagen das ECDL-Konzept fördert Unwissen bzw. Funktionenwissen; die Objekt-Attribut-Methode fördert Wissen bzw. Hintergrundwissen und Verständnis.

Der ECDL erklärt lediglich einige Funktionen, dieses kann jedoch nicht als Wissen betrachtet werden. Die Struktur der Programme bei gänzlich im Unklaren. Damit weiß man nicht mehr als wie im Ein-

zelfall eine bestimmte Sache machen kann. Darüber hinaus kann man damit nichts weiteres anfangen. Es ist für den einzelnen nicht nachzuvollziehen, warum eine Sache mit dem Computer in dieser bestimmten Weise ausgeführt werden soll. Zudem dürfte es für jemanden, der noch keine Ahnung von Computern hat schwierig sein, dieses "Wissen" auf Dauer zu behalten. Da er keine andere Möglichkeit aufgezeigt wird, als es auswendig zu lernen. Mit jeder neuen Version von einem Softwarepaket könnte das Gelernte unnütz werden, da die einzelnen Funktionen verändert wurden. Es ist nicht übertragbar.

Die Objekt-Attribut-Methode ist das genau Gegenteil dazu. Hier wird ein Methode vermittelt, die die Struktur von Programmen offen legt. Es wird nicht eine einzelne Funktion eines Programms erklärt, sondern ein Methode, mit der man sich die Funktionsweisen von Programmen erklären kann. Sie gewährt auch schon einen kleinen Einblick in informatische Konzepte. Mit dem Verständnis kann man dann weit mehr als nur ein spezifisches Programm zu bedienen. Es ist weitaus leichter zu erlernen, da der Kern nur eine Methode ist, die man auf nahezu jedes Programm anwenden kann. Damit ist sie unabhängig von einer konkreten Softwareprodukt. Es wird den Schüler damit ermöglicht, weiter informatische Bereiche sowie andere Anwendungsprogramme zu erschließen.

Der Unterschied von Wissen und Unwissen liegt, meiner Meinung nach, in einer erklärten Sache, also ihre Struktur und Eigenschaften kennen und ihre Zusammenhänge verstehen, und der nicht Nicht-Erklärung selbiges. Wie man die Sache dann benutzen kann, hängt von ihren Eigenschaft ab bzw. ergibt sich aus diesen. Der ECDL leistet dies nicht.

Didaktik der Informatik II
Blatt 3

Aufgabe 1)

Eine Begründung für die Objekt-Attribut-Methode ist, dass durch die Nutzung von Anwendungsprogrammen auch deren Funktionsweise und Bedienung erlernt wird, wodurch die Schüler sicherer im Umgang damit werden. Nach einer Studie finden sich 41% der Schüler nicht ausreichend qualifiziert, wo diese Methode gegen wirkt. Eine weitere Begründung liegt im realitätsnäheren Bezug der Objekte und Attribute, da die Schüler direkt damit arbeiten und sie nicht so abstrakt wie bei Java sind. Weiterhin wird so dem falschen Bild der Informatik vorgebeugt, dass Informatik nur aus Programmieren besteht.

Bsp. 1:

Textverarbeitung:

Klasse: Schrift

Objekt: eine bestimmte Schrift, z.B: Arial

Attribute: Größe, Farbe, Fett, Kursiv, Unterstrichen...

Methoden: Kopieren, Verschieben, Einfügen, Vergrößern...

Bsp. 2:

Zeichenprogramm:

Klasse: Linie

Objekt: Gerade Linie

Attribute: Länge, Dicke, Farbe, Richtung

Methoden: Verschieben, Zeichnen, Verlängern, Richtung ändern

Bsp. 3:

Mp3-Player:

Klasse: Lied

Objekt: bestimmtes Lied

Attribute: Länge, Bitrate, Mono, Stereo, Größe

Methoden: Abspielen, Stoppen, Vor-, Zurückspulen, Löschen

Aufgabe 2)

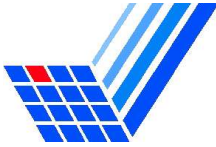
Als Vorteile sehe ich den näheren Realitätsbezug für die Schüler, da sie öfter mit Anwendungsprogrammen arbeiten und so sehr wahrscheinlich etwas motivierter sind. Ebenso wird durch das Vermitteln des Verständnisses, was in Anwendungsprogrammen zusammen gehört, die Funktionsweise der Programme deutlich und die Schüler können sich schneller in unbekannte Programme einarbeiten. Weiterhin wird so nicht die Informatik als „Programmierwissenschaft“ gesehen, aber auch nicht der richtige Blick auf die Informatik aufgezeigt. Nachteilhaft ist meiner Meinung nach, dass die Schüler sehr wenig über die Wirkungsweisen unter dem Programm erkennen, da sie nur auf der Benutzeroberfläche arbeiten und die genauen Funktionen nie kennen lernen. Allerdings denke ich, dass der Nutzen auf andere Dinge doch vorhanden ist. Sollten die Schüler die objektorientierte Sichtweise verstanden haben, ist es leichter, eine Abstraktionsstufe höher zu gehen und in Java z.B. das Wissen anzuwenden und dann auch dort leichter zu verstehen, warum es so gemacht wird.

Aufgabe 3)

Die beiden Konzepte haben es gemeinsam, dass in beiden vermittelt wird, wie Anwendungsprogramme bedient werden. Aber das war auch schon fast alles gemeinsame. Im ECDL wird dies hauptsächlich durch Auswendiglernen, oder durch Vorgabe der Aktionen der Programmablauf erklärt, welcher dann auswendig für die Prüfung gelernt werden muss. In der Objekt-Attribut-Methode hingegen, werden die verschiedenen Möglichkeiten der Bearbeitung in zusammenhängende Module gefasst und die Zusammenhänge erläutert, die sie auch übertragbar auf neue Programme machen. Diese neue Methode geht also etwas mehr auf das Warum ein und erklärt Zusammenhänge.

Aufgabe 4)

Ich bin der Meinung, dass die Objekt-Attribut-Methode sinnvoller ist. Sie geht über das einfache Anwenden hinaus und versucht, die Funktionsweisen der Anwendungsprogramme zu erklären. Da die Schüler sich dabei Gedanken machen müssen, ob und wie verschiedene Sachen zusammengehören, ist der Lerneffekt größer, als bei dem ECDL, der nur auf eine Prüfung abzielt. Ich finde, wer die Prinzipien verstanden hat, wie die Anwendungsprogramme aufgebaut sind ist eher berechtigt, so einen Computerführerschein zu bekommen, als jemand, der das machen kann, was ihm vorgeführt wurde. Was mir allerdings bei der Methode fehlt ist der Aspekt der Datensicherheit und des Datenschutzes.



Übung zur Vorlesung „Didaktik der Informatik II“

Blatt 4

In der Vorlesung wurde im Zusammenhang eines internationalen Vergleichs von Informatikkompetenzen (Stichwort: PISA-Studie) ein Artikel von Hermann Puhmann¹ 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 an, wie der Informatikunterricht in der Sekundarstufe II auf solcherart veränderte Eingangsbedingungen reagieren kann/sollte. Beschreiben Sie dazu einen unterrichtlichen Rahmen (inhaltlich und methodisch) für den Anfangsunterricht in der Sekundarstufe II, der davon ausgeht, dass die o.g. Kompetenzen vorhanden sind.

Abgabe: 12.11.2003, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ Hermann Puhmann: 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 Lehreranstellung „Didaktik der Informatik II“ aufgeführt.

Ausgewählte Antworten zur 2. Aufgabe:

Aufgabe 2:

Wenn diese Kompetenzen vorhanden sind, kann davon ausgegangen werden, dass Grundlagenwissen im Bereich der Informatik vorhanden ist. Es ist bekannt, wie Programme anzuwenden sind und wie Informatiksysteme dem Grunde nach funktionieren.

Auf diesem Wissen kann aufgebaut werden: Ausgehend von dem Wissen über die Grundlagen der Struktur von Informatiksystem, kann man dieses vertiefen. Dazu kann man z.B. Anwendungsprogramme, über die Wissen der Grundstruktur existiert, benutzen, um tieferliegende Strukturen, wie die von internen Datenstrukturen und Modellen der Systeme zu erläutern. Weitergehender können dann am Beispiel einfachere Algorithmen, die in den System vermutlich benutzt werden, erläutert werden.

Es muss nun, im Informatikunterricht kein Wissen über die Anwendung von Programmen mitvermittelt werden. Dieses Wissen kann in sofern als Grundlage für den Unterricht benutzt werden, als das Funktionsweisen, die bekannt sind, als Ausgangspunkt genommen werden, um daran informatische Methoden, Modelle und Algorithmen zu erklären. Der Unterricht sollte also da ansetzen, wo das Wissen der Schüler aufhört.

Die Schüler verfügen nun zwar über grundlegendes Wissen der Struktur von Informatiksystemen, diese ist jedoch eher anwendungsorientiert zu sehen. Sie kennen Strukturen der Systeme, die Auswirkungen auf die Anwendungen haben bzw. können Auswirken erklären. Der Ansatzpunkt kann hier ein Ansichtsseitenwechsel von Informatiksystemen sein. Bekannt ist wie Systeme modelliert sind bzw funktionieren, aber noch nicht (oder nur in Grundzügen) wie man diese Modellierung selbst vornimmt.

Es können also z.B. Anwendungsprogramme hergenommen werden, und Modelle von ihm erstellt werden: Wie ist eine graphische Oberfläche modelliert; gibt es Möglichkeiten der Verbesserung von ihr. Wie sind die Daten der Anwendung modelliert, können dort Verbesserungen gemacht werden und was ist nötig damit diese Daten andere Anwendungen und Systeme benutzen können.

Dieses kann fortgesetzt werden, bis hin zu Algorithmen auf Datenmodellen. Darauf hin könnte man dann kleinere Informatiksysteme die Schüler selbst entwickeln lassen.

Auch alternative Modellierungsmethoden können an bestehenden Systemen thematisiert werden.

2. Aufgabe

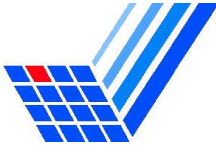
Sind beim Eintritt in die Sek II die oben genannten und weitere Kompetenzen voraussetzbar, bilden sich für den Unterricht ganz neue Möglichkeiten. Heute wird häufig gerade zu Beginn der Sek II während des ersten (Halb-)Jahres Grundlagenwissen erarbeitet (natürlich nur im Schnelldurchgang) oder aufgefrischt. Auch im späteren Verlauf muss immer wieder durch Wiederholungen oder thematische Exkursionen nachgeholt werden, was in der Sek I nicht geleistet wurde oder werden konnte.

Selbsterklärend wird dadurch wertvolle Unterrichtszeit für die intensivere Behandlung der geforderten Inhalte oder für neue Inhalte frei. Natürlich muss die Methodik im Unterricht weiterhin Hilfestellungen bei Fragen bezüglich des Stoffs bieten und sollte niemals "blind" auf die erfolgreiche Aneignung der Kompetenzen vertrauen.

Einen weiteren Vorteil sehen ich darin, dass Schüler zusammen mit den Kompetenzen das Abstrahieren oder Modelldenken früher lernen. Klar, jedes Beispiel hinkt, aber man bräuchte nicht einem Sek II-Schüler Schritt-für-Schritt-Anweisungen zu geben, wie er in Word die Schriftfarbe verändert. Er würde höchstwahrscheinlich über die Information verfügen: "In jeder ordentlichen Textverarbeitung muss ich doch die Schriftfarbe ändern können" und dann unabhängig vom Programm auf Suche gehen können.

Besonders was Prüfungen, beispielsweise Klausuren, angeht, steht der Lehrer oftmals vor einem großen Kompetenzgefälle unter den Kursteilnehmern, da kaum eine Frage aus dem Informatikunterricht ohne zusätzliches Hintergrundwissen sehr gut beantwortet werden kann. Durch die gewonnen Zeit wird es ferner möglich themenübergreifenden Unterricht durchzuführen, der sich nicht auf die minimale Abhandlung der Richtlinien beschränkt, sondern auch Bezüge zwischen Themengebieten herstellt. Ein Beispiel wäre hier die Verknüpfung der in Aufgabe 1A erwähnten Suchmaschinen mit Datenbanken und Internetseiten(aufbau) (Leitfrage: Wie speichert die Suchmaschine ihre Daten?). Die Unterschiede bei pixelorientierten / vektororientierten Zeichenprogrammen können sehr gut anhand der zugrundeliegenden Datenstrukturen zur Speicherung der Zeichnung verknüpft werden (besonders schön und zurecht beliebt: objektorientierte Datenhaltung in Verbindung mit Vektorgrafiken). Auch der oft verlangte fächerübergreifende Unterricht ist nur möglich, wenn in jedem der zu verknüpfenden Sachgebiete Grundlagenwissen über spezifische Vorgehensweisen und Modelle bekannt sind. Je früher daher solche Grundkompetenzen erworben werden, umso besser.

Gerade das ist ja auch ein Ziel des Projektunterrichts. Oftmals kommt jedoch gerade in der Sek II der Projektunterricht wegen Zeitmangel zu kurz. Da heutzutage nie vorhersehbar, wie lange die Eingangsphase der Sek II für einen Informatik-Kurs andauern wird, kann auch nicht von vorneherein sinnvoll eine Projektarbeit geplant werden. Sind hingegen die Kompetenzen im großen und ganzen voraussetzbar, dann läßt sich so etwas hervorragend vorbereiten und bewerkstelligen.



Übung zur Vorlesung „Didaktik der Informatik II“

Blatt 5

Eine Möglichkeit, allgemeinbildenden Informatikunterricht für Schülerinnen und Schüler motivierend zu gestalten besteht darin, bei der Planung von realweltlichen Problemen auszugehen. Dabei können auch Themen, die auf den ersten Blick nicht unbedingt als Informatikproblem angesehen werden, aufgegriffen werden.

1. Aufgabe: realweltliche Probleme (2 Punkte)

Zählen Sie eine Reihe von realweltlichen Problemen zum Thema „Handy“ auf, mit denen Schülerinnen und Schüler in ihrem Alltag konfrontiert sein können. Es sollten zu diesem Zeitpunkt möglichst konkrete Probleme formuliert werden, die als Ausgangspunkt eines problemlösenden Informatikunterrichts dienen können.

2. Aufgabe: Problemstellungen (2 Punkte)

Wählen Sie zwei Ihrer konkreten Probleme, durch die Schülerinnen motiviert werden können, sich mit den im Zusammenhang auftretenden *informatischen* Fragestellungen zu beschäftigen. Formulieren Sie zwei Problemstellungen nach dem im Skript angegebenen Muster¹.

3. Aufgabe: Konkretisierung der thematischen Orientierung (2 Punkte)

Konkretisieren eine der beiden Problemstellung zu einer Unterrichtsplanung, indem Sie jeweils mögliche Methoden, Inhalte, Sozialformen und Medien beschreiben.

4. Aufgabe: Modulkonzept der informatischen Bildung (2 Punkte)

Stellen Sie für Ihre Unterrichtsplanung dar, wie die drei Dimensionen des Modulkonzepts der informatischen Bildung berücksichtigt werden können.

5. Aufgabe: Lehrplan (2 Punkte)

Zeigen Sie Bezüge zum Informatik Lehrplan NRW² mit Angabe der Seitenzahlen und Kapitelüberschrift auf.

Abgabe: 19.11.2003, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ siehe Skript zur Vorlesung „Didaktik der Informatik 2“, Kap. 5

² Bitte lesen Sie im Skript zur Vorlesung „Didaktik der Informatik 1“, Kap. 7, wie Sie eine lokale Kopie des Lehrplans erhalten können.

Didaktik der Informatik II -- Lösung zu Blatt 5

Aufgabe 1:

- * Kurzmitteilungen kommen beim Empfänger nicht (oder viel zu spät) an, obwohl das Handy des Absenders den korrekten Versand bestätigt hat, und das Handy des Empfängers eingeschaltet ist.
- * Kurzmitteilungen kommen beim Empfänger an, obwohl das Handy des Absenders den korrekten Versand nicht bestätigt hat.
- * In den Menüstruktur eines Handys, gibt es unterschiedliche Menüs für die selben Dinge, die auf verschiedenen Wegen durch die Menüstruktur erreicht werden. (Z.B. gibt es für Telefonbücher bei manchen Handys mehrere Menüs)
- * City/Home-Tarif funktionieren nicht in Bereichen in denen sie eigentlich funktionieren sollten.
- * Ein Handy schaltet sich ohne ersichtlichen Grund einfach ab. (Firmware Probleme)
- * Verbindungen brechen ab, während man sich mit einem Zug, Auto o.ä. bewegt. (Funklöcher)
- * Handy haben störende Auswirkungen auf andere Elektrogeräte, wie Radios, Bildschirme usw.

Aufgabe 2:

Problemstellung Kurzmitteilungen:

Sabine und Marion haben beide seit einiger Zeit ein Handy. Um Telefongebühren zu sparen, verabreden die beiden sich häufiger per Kurzmitteilungen. Von Zeit zu Zeit kommt es jedoch vor, obwohl beide ihr Handy eingeschaltet haben und die Mitteilung laut Handy der Absenderin korrekt Versand wurde, dass eine von den beiden die Kurzmitteilung mit dem Termin

- * erst kurz vor der Verabredung erhält,
- * gar nicht erhält oder
- * erst nach der Verabredung erhält.

Wie können die beiden dieses Probleme verhindern, ohne jeweils eine Bestätigungsmittteilung zu verschicken, die zusätzlich Geld kostet?

Problemstellung City/Home-Tarif: Siehe dazu: Harald Bögeholz, Dusan Zivadinovic: Telefon-Zellen, in c't 18/99, S. 174, [<http://www.heise.de/ct/99/18/174/||c't 18/99>]

Vor kurzem hat Sabine einen Handyvertrag bei einem bekannten Mobilfunkunternehmen abgeschlossen. Da sie zu Hause keinen eigenen Telefonanschluss hat und ungestört, ohne die lästigen Beschwerden ihrer Eltern wegen der Telefongebühren, von ihrem Zimmer aus telefonieren will, wählt sie einen Vertrag mit einem Home-Tarif aus, damit sie von zu Hause günstiger telefonieren kann. Doch leider musste sie feststellen, dass sie in ihrem Zimmer nicht zu dem günstigeren Tarif telefonieren kann; im Rest der Wohnung ihrer Eltern geht dies jedoch einwandfrei.

Was soll bzw. kann Sabine nun tun und woher rührt ihr Problem?

Aufgabe 3:

Die oben beschriebenen Problem mit den Kurzmitteilung sollte wohl jeder Schüler, der ein Handy besitzt, kennen. Auch die Lösung des Problems dürfte bekannt sein -- es gibt keine. Vermutlich können die Schüler jedoch nicht beantwortet, warum diese Probleme auftreten.

Zur Klärung dieses Problems ist es notwendig, dass der Aufbau des Kurzmitteilungsdienstes (SMS) erläutert wird. Es handelt sich dabei um ein Client-Server-System, wobei die Handys jeweils Clients darstellen und die Kurzmitteilungszentrale -- den Server. So bietet es sich an, auf solche System so wie auf Protokolle zwischen Clients und Server im allgemeinen (theoretischen) näher einzugehen. Der SMS kann dafür als Grundlage dienen.

Bei der Erläuterung des Dienstes sollte darauf geachtet werden, dass man sich nicht in Details verstrickt, da es dabei nur wichtig, ist wie der Dienst schematisch funktioniert, d.h. grob die Nachricht wird an einen Server übermittelt, das Handy des Absenders erhält über die erfolgreiche Übermittlung an den Server eine Nachricht, die Nachricht wird an das Handy des Empfängers übermittelt oder gelöscht falls dieser in einen bestimmten Zeitrahmen nicht erreichbar ist. Letztendlich sollte man zu einer Beurteilung des Dienstes kommen: Das Problem ist schon in der Modellierung des Dienstes angelegt.

Weitergehend kann man auf Standards eingehen, da der SMS ein Teil des GSM-Standards ist. Als Leitfragen sind zu nennen: Was für Arten von Standards gibt es? Wofür und warum werden Standards benötigt? Von wem oder wie werden Standards festgesetzt oder durchgesetzt? Viele Quasi-Standards haben sich einfach durch ihre Benutzung durchgesetzt und gewisse Firmen haben da noch so ihre ganz eigenen Methoden zur Durchsetzung von Standards...?

Die weitere Beantwortung stellt sich für mich schwierig dar, da die zu Nutzen Methoden, Sozialformen und Medien stark von der Lerngruppe und von den örtlichen Gegebenheiten abhängen.

In jedem Fall sollte der Struktur des Problems gefolgt werden: Zunächst die Erläuterung des Dienstes als Client-Server-System. Darauf hin kann ein Übergang zu allgemeinen Client-Server-System gemacht werden, womit eine Bewertung des Dienstes möglich wird. Weiterhin kann dann auf den Dienst als Teil eines Standards eingegangen werden. Letztendlich können dann Standards im allgemeinen besprochen werden. Diese Struktur ist notwendig, damit die Schüler den Zusammenhang zum Problem erkennen können.

Für die Darstellung von Client-Server-System bieten sich Sequenzdiagramme an. Der Zugang zum Problem kann, durch Sammeln der Schüler Erklärungs(ansätze), erreicht werden. Es können weiter Systeme, sofern in der Schule ein Netz zur Verfügung steht, wie Mail-Systeme, verwendet werden, bei denen ein der Datenaustausch zwischen Client und Server mitprotokolliert werden kann. Zu dem Standardisierungs-Bereich können die Schüler in Gruppenarbeit im Internet recherchieren. Auch die Modellierung eines anderen Kurzmitteilungsdienstes durch die Schüler könnte in Gruppenarbeit bewerkstelligt werden. Des weiteren können sicherheitsrelevante Aspekte von Netzwerken beleuchtet werden.

Aufgabe 4:

Informatiksysteme verstehen und verantwortlich nutzen

Hierzu kann auf das Thema Sicherheit in Netzwerken Bezug genommen werden. Dies ist gerade bei Handys ein wichtiges Thema, da hier finanzielle Nachteile bei dritten entstehen können. Auch das Mithören von Gesprächen kann Berücksichtigung finden.

Bei den Kurzmitteilungen ist es auch wichtig zu wissen, dass den Absender nicht immer die Schuld am verspäteten oder nicht Erhalten der Nachricht trifft. Dieses ist zwar nicht eins zu eins auf andere informatische Systeme zu übertragen; es ist aber zu beachten, dass bei der gesellschaftlichen Benutzung von Informatik-Systemen Probleme entstehen können, die nicht immer auf die Anwender zurückzuführen sind, sondern System bedingt sein können.

Modellierung -- zentrales Feld informatischer Arbeit

Hierzu kann die objektorientierte Modellierung genannt werden. Bei Protokollen bieten sich Sequenzdiagramme an, die ein Teil der OOM sind. Es können von den Schülern auch selber Protokolle modelliert werden, die alternative zu dem bestehenden sind. Dieses Modell kann ersatzweise (man wird wohl keine Möglichkeit haben, es in Hardware zu implementieren) auf einen Computer implementiert werden.

Erkenntnisse der theoretischen Informatik im Anwendungskontext

Im Unterricht kann auch allgemein auf Client-Server-Systeme eingegangen werden. Dabei können Protokolle und Sicherheitsmaßnahmen betrachtet werden. So das auch dieser Bereich ausreichende Betrachtung findet.

Dabei kann auch Bezug auf Standardisierungsverfahren (Methoden, Formale Sprachen usw.) genommen werden.

Aufgabe 5:

Bezüge mit dem Lehrplan ergeben sich hier im Besonderen mit folgenden Teilen des Lehrplans:

- * 2.2.1.1 Modellierung und Konstruktion S. 11ff
- * 2.2.1.2 Analysieren und Bewerten S. 14ff
- * 2.2.2 Lernen im Kontext der Anwendung S. 16ff
- * 2.2.3 Methoden und Formen selbstständigen Arbeitens (inkl. Unterabschnitte) S. 20ff
- * 2.3.3.2 Objektorientierter Ansatz S. 29ff

Diese ergeben sich daraus, dass ein konkretes System der Informatik zunächst analysiert. Dazu werden objektorientierte Methoden verwandt. Weiterhin sollen die Schüler eigenständig ein alternatives System entwickeln. Bei der Analyse können Themen andere Fächer berührt sein, da sich hier um Handy handelt, etwa die der Physik. Außerdem werden theoretische Grundlagen am konkreten Problem behandelt. Womit dann zu einer Bewertung des Anfangs analysierten Systems gelangt wird.

Blatt 5:

1) Realweltliche Probleme:

- Es kommt vor, dass man überhaupt keine Verbindung mit dem Handy herstellen kann, obwohl zu anderen Zeiten vom selben Ort aus eine hervorragende Signalstärke erreicht wird. Warum?
- Manchmal reißt eine Verbindung mitten im Gespräch ab, oder man hört den Gegenüber gar nicht mehr. Warum?
- Handys stürzen bekanntlich ab. Wieso können Handys überhaupt abstürzen?
- Immer häufiger bekommt man Kurznachrichten, die zum Wählen irgendwelcher 0190-Nummern auffordern um Gewinne einzufordern. Was kann man dagegen tun?
- Java-Software (Spiele, Programme) funktioniert nicht auf allen Handys. Wieso ist das so?
- Eine verschickte SMS sieht bei dem Empfänger ganz anders aus, als beim Versender. Warum?

2) Problemstellungen

Zum Problem Sechs (Problem Fünf könnte ähnlich formuliert werden...):

a) Eine SMS kann bekanntlich Sonderzeichen, mehrere aufeinander folgende Leerzeichen, Umlaute, u. ä. enthalten. Wird nun eine derartige Nachricht an ein Handy in einem andern Land und oder an ein Handy eines anderen Herstellers versendet, kann es vorkommen, dass:

- Zeilenumbrüche verloren gehen,
- Mehrere Leerzeichen zu einem Einzigen zusammengefasst werden,
- ASCII-Grafiken wegen fehlender Leerzeichen falsch dargestellt werden,
- Umlaute "weggelassen" werden,
- Sonderzeichen falsch oder überhaupt nicht angezeigt werden.

Was sind die Ursachen, und wie kann man das Problem umgehen?

Ist zwar ähnlich wie im Skript, aber eben im Bereich Handy und um ein paar wie ich finde wichtige Punkte bereichert:

b) Ohne jemals seine Handynummer im Internet oder einer fremden Person gegenüber bekannt gemacht zu haben, bekommt man mit der Zeit immer häufiger Kurznachrichten von unbekanntem Personen oder Institutionen, die

- dazu auffordern eine 0190* (oder 0900*) -Nummer zu wählen
- dazu auffordern, eine Kurznachricht an eine bestimmte Rufnummer zu senden
- um eine Bestätigung der Kundendaten und -Passwörter bitten
- das Handy zum "Absturz" bringen
- einfach nur lästig sind

Wie sollte man reagieren?

3) Konkretisierung der thematischen Ordnung

Zum Punkt 2b) hier eine tabellenartige Aufstellung der Unterrichtsplanung:

a) Einführung:

- Einschlägige Erfahrungen der SchülerInnen sammeln, wann, wie häufig und mit welchem Inhalt Spam-SMS den Weg auf ihr Handy finden / offenes Unterrichtsgespräch / als Material werden entsprechende Beispiel-Nachrichten gezeigt (TLP, o. ä)
- Erklärung des Begriffs "Spam" mit Veranschaulichungen aus der Welt der E-Mails / analog: Spam-Mails als Material 7 Frontalunterricht
- Ergründung und Gruppierung der Absichten und Ziele von Spam(-Versendern) ("Was will den der andere von mir?") / Gruppen- oder Partnerarbeitsphase /
- Gefahren die daraus resultieren (Kosten, Identitäts-Diebstahl, etc) / Gruppen- oder Partnerarbeitsphase / evtl (falls verfügbar) könnten die Schüler im Internet mit den erarbeiteten Zielen der Spammer im Hinterkopf, nach Erfahrungsberichten suchen, die die Folgen schildern. / Medien: Einschlägige Nachrichten aus Fernseh-Reportagen, Zeitungsausschnitte

b) Prävention & Abwehr:

- Klärung der Frage, wie der Spammer an die Handynummer des Opfers kommt; Überlegungen, wo überall die persönliche Handynummer auftaucht. / Materialien: Dokumente, die Handynummern erwarten. Beispielsweise Umfragebögen, Treupunkte-Karte, etc. / Methodik:
- Reaktion auf Handy-Spam, Welche Reaktionen sind vorteilhaft, welche nicht? / offenes Unterrichtsgespräch und evtl. Rollenspiel
- Filtertechniken (rufnummernbasiert, inhaltsbasiert) und deren Grenzen. / Unterrichtsgespräch / Material: false Positives SMS und nichterkannte Spam-SMS für unterschiedliche Vorschläge bereithalten um auf die Grenzen hinzuweisen

c) Datenschutz

- Feststellung der offiziellen Datenschutzbestimmungen / Gruppenarbeit evtl vorbereitetes Referat
- Prüfung der tatsächlichen Einhaltung der Datenschutzbestimmungen / Methode: Internetrecherche als Partnerarbeit
- Schlupflöcher, Konstruktion von Fällen, die durch die Datenschutzbestimmungen nicht abgedeckt werden können oder dürfen / diverses Arbeitsphasen oder Hausaufgaben

d) Zukunftsaussichten:

- Modellierung eines "spam-sicheren" Mobilfunknetzes mit unterschiedlichen Sicherheitsstufen, Rückverfolgung, etc / Projektarbeit

4) Modulkonzept der informatischen Bildung

Modulbezeichnung:

Ich nenne hier die relevanten Bereich aus Aufgabe 3

- Informatiksystem verantwortlich nutzen: a, b

Hier werden die Schüler für den "verantwortlichen" das schließt ja "vorsichtigen" Gebrauch des Informatiksystems "Handy" geschult. Gefahren zu laxer Handhabung persönlicher Daten oder des naiven Befolgen von Anweisungen Unbekannter sollen aufgezeigt werden. Die Bedeutung der Spam-Werbung unter wirtschaftlichen Aspekten (sowohl für den Auftraggeber als auch für den Geschädigten) muss aufgezeigt werden mit dem Ziel der Sensibilisierung und der verantwortlichen Nutzung dieser (heutzutage sogar teilweise kostenlosen) Dienste

- Modellierung: c, d

Unter dem Teil c sollen ja Schlupflöcher und Konfliktsituationen in den aktuell (ehrlich gesagt wird es die in jeder Datenschutzbestimmung geben) geltenden Datenschutzbestimmungen (theoretisch) modelliert werden, um die Grenzen derselben aufzuzeigen. Es kann gezeigt werden, dass die Informatik eben nicht jedes Problem lösen kann. Im Teil d schließlich soll (zur Vereinfachung) die reine SMS-Kommunikation in einem Mobilfunknetz derart modelliert werden, dass man es als "spam-sicher" bezeichnen kann. Hierzu werden die SchülerInnen geeignete Algorithmen, Datenstrukturen und Sicherheitsprinzipien kennen oder erarbeiten müssen.

- Theoretische Informatik: b, d

Wie eben schon erwähnt, werden besonders in der Projektarbeit (Teil d) die Erkenntnisse aus der Theoretischen Informatik gute Anwendung finden. Im Teil b (speziell Filtertechniken) soll schon vorgehend darauf eingegangen werden. Zur Frage, wie der Spammer an die Handynummern der Opfer kommt, können ebenso theoretische Kenntnisse zur Datenverarbeitung und -übermittlung herangezogen werden.

5) Lehrplan

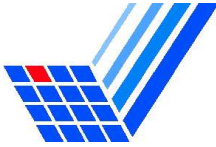
Hier folgt eine Gegenüberstellung der Lehrplan-Inhalte zu der Unterrichtsplanung aus Aufgabe 3:

2.2.1.1 Modellieren und Konstruieren - c (Schluplöcher), d

2.2.1.2 Analysieren und Bewerten - a, b, c

2.2.3 Methoden und Formen selbständigen Arbeitens - a, b, c (jeweils immer mal wieder während den Partner-/Gruppenarbeitsphasen), d (Projektarbeit)

2.2.3.3 Fächerübergreifende Sichtweisen - c (in Verbindung mit z. Bsp. Politik)



Übung zur Vorlesung „Didaktik der Informatik II“

Blatt 6

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 den Unterricht in der Schule 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 in der Sekundarstufe II. Fassen Sie Ihre Ergebnisse sehr knapp tabellarisch zusammen und geben Sie gegebenenfalls stichwortartig an, warum es nicht geeignet ist.

2. Aufgabe:

(3 Punkte)

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

3. Aufgabe:

(2 Punkte)

Adaptieren Sie die Übersichtskarte für die Sekundarstufe II, indem Sie Ihre Ergebnisse ggf. umgruppieren, neue Muster hinzufügen und unrelevante Muster entfernen.

Abgabe: 26.11.2003, 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 paradigmenergreifenden Pizza-Beispiel. An dieser Stelle folgen einige Überarbeitungen und Neuentwürfe von Mustern.

Aufgabe 2:

Angepasste Patterns

Lass die Schüler entscheiden (3. Let Them Decide)

Die Schüler sollten die Möglichkeit haben, den Unterricht mitzugestalten. Da die Schüler sich nur wenig mit Informatik auskennen, können sie die Themen nicht vollständig selber auswählen. Es können mit den Schülern zusammen jedoch Themenkomplexe erstellt werden. Die Interessen der Schüler sollten hier bei berücksichtigt werden. So kann man die Schüler nach ihren Interessen und Problem mit Informatiksystem fragen, und daraus zusammen mit den Schülern Unterrichtssequenzen entwickeln.

Themen in Komponenten zerlegen (29. Digestible Packets)

Ein Thema wird in der Schule häufig über einen längeren Zeitraum unterrichtet. Eine Stunde sollte jedoch in sich abgeschlossen sein. Daher ist es wichtig, die Themen zu untergliedern. Dabei ist auf die Abgeschlossenheit jedes Gliederungselementes zu achten. Wenn Unterthemen zu groß werden, wird es für die Schüler schwierig zu folgen.

Wiederholungen (33. Repeat Topics)

Wiederholungen von Themen führt zu Verbesserung des Lernerfolgs. In der Schule können jedoch nicht die kompletten Themen wiederholt werden. Es ist jedoch sinnvoll, die wichtigsten Aspekte eines Themas an passenden Stellen zu wiederholen. Dabei sollte darauf geachtet werden, dass dieses kurz und präzise passiert, weil sich die Schüler ansonsten langweilen.

Neue Patterns

Klassenarbeiten bzw. Klausuren und Test einplanen

Mehrere Klausuren werden in der Schule während des Halbjahres geschrieben. Häufig passiert es, dass dies in der Planung nicht beachtet wird. Dann muss kurz vor der Klausur entsprechender Stoff schnell behandelt werden. Bei schneller Behandlung eines Themas besteht die Gefahr, dass die Schüler diesen nicht verstehen.

Deswegen beachte bei der Planung, dass Klausuren geschrieben werden müssen. Dabei ist darauf zu achten, dass genug Material zum Zeitpunkt an dem die Klausur geschrieben wird behandelt worden ist. Eine Klausur sollte mit dem Themen auch nicht überfrachtet werden. Außerdem sollte darauf geachtet werden, dass ausreichend Zeit für die einzelnen Themen bereit steht.

Um dieses zu erreichen, sollten die Themen am Anfang des Halbjahres feststehen. Dabei sollte beachtet werden welche Themen für die Klausuren benutzt werden sollen. Stellt sich hinterher heraus, dass die Zeit für ein Thema vor einer Klausur nicht mehr ausreichend ist, sollte es besser nicht mehr in der Klausur Verwendung finden. Letzteres sollte man den Schülern auch mitteilen.

Referate

Besser gelernt wird, wenn sich ein Schüler oder eine Gruppe von Schülern selbst mit einem Thema beschäftigt. Deswegen können auch Referate ein sinnvolles Mittel des Unterrichts sein. Der bzw. die Schüler müssen sich dann intensive mit einem Thema befassen. Außerdem kann so die freie Rede von Schülern geübt werden.

Hierfür sollte eine Reihe von Themen bereit stehen, an den die Schüler Interesse gezeigt haben. Auch das Material für dieses Thema sollte bereit gestellt werden. Bei der Auswahl der Themen sollte darauf geachtet werden, dass sie vom Umfang in etwa gleich sind und der Schwierigkeitsgrad variiert, damit für Schüler verschieden Leistungsstandes entsprechende Themen bereit stehen. Auch ist darauf zu achten, dass die Themen von Interesse für die Schüler ist.

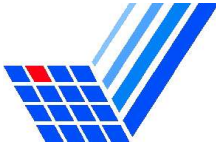
Die Themenauswahl kann von den Schülern selbst erfolgen. Der Vorteil darin liegt, dass die Schüler dann offensichtlich ein Interesse an ihnen haben. Die Referate sollten nicht länger als 10-15 Minuten dauern, damit der Arbeitsaufwand für die Erstellung des Referats angemessen bleibt und die Zuhörer sich nicht langweilen. Für die Zuhörenden Schüler ist es wichtig, dass nicht mehr als ein Referat pro Stunde gehalten werden sollte, da ansonsten schnell Langweile eintritt. Im Rest der Zeit kann das Thema des Referats weitergehender behandelt werden. Der Lehrer sollte für Fragen die bei der Ausarbeitung des Referats entstehen, bereit stehen und dieses den Schülern auch mitteilen, denn oft haben Schüler Angst, sich beim Referat zu blamieren.

Projekte

Durch intensive Behandlung eines Themas mehrere Schüler, wie das in Projekten der Fall ist, kann das Verständnis der Schüler für das Thema verbessert werden. Die Beschäftigung der Schüler selbst kann den Lernerfolg verbessern. Außerdem ist die geleistete Arbeit der Schüler als Produkt des Projektes unmittelbar sichtbar. Die Schüler sehen was sie gelernt haben und wofür man es benutzen kann.

Hierfür können verschiedene Themen als auch ein Thema bereit stehen. Die Auswahl der Themen bzw. des Themas sollte hier auch den Schülern überlassen werden. Es können verschiedene Gruppen verschiedene Themen behandeln also auch verschiedene Projekte zu einem Thema möglich sind. Die Ergebnisse können dann in kurzen Referaten von den Schülern vorgetragen werden.

Hier können mehrere Wege gewählt werden. Projekte können sich als Übung an ein Thema im Unterricht anschließend. Die Projekte sollen von dem Schülern ausgewählt werden können, damit das Interesse der Schüler erreicht wird. Das Projekt sollte hauptsächlich im Unterricht selbst stattfinden so, dass die Schüler auch die Möglichkeit haben auftretende Fragen zustellen. Der Lehrer sollte beim Projektablauf den Schülern helfend zur Seite stehen. Bei Projekten zu mehreren Themen sollten die Schüler nach Beendigung in Referaten die Ergebnisse vortragen. Auch das Vortragen von Zwischenergebnissen kann sinnvoll sein. Die Materialien der Projekte sollen hier auch von Lehrer bereit gestellt werden. Bei Projekten zu einem Thema kann/sollte der Lehrer selbst auch ein eigenes "Referenz Projekt" erstellen.



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

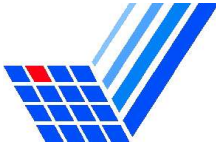
Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

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 II“

Blatt 8

In der Vorlesung wurde das Buch „How to Think Like a Computer Scientist – Learning with Python“ von Gregor Lingl⁷ 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 II zu entwickeln.

1. 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 mögen⁸. Da 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)

2. Aufgabe:

(3 Punkte)

Entwickeln Sie analog zu Aufgabe 1 ein anderes Beispiel als das Kartenspiel zum Kapitel 15, „Mengen von Objekten“. Ähnlich wie in dem genannten Beispiel sollten die Objekte zumindest miteinander verglichen werden können, hinzugefügt und entfernt werden können sowie auch ausgegeben werden können. Finden Sie auch bei diesem Beispiel motivierende Aufgabenstellungen aus dem Umfeld der Schülerinneninteressen.

Hinweis: Sie brauchen für die 2. Aufgabe kein Programm zu implementieren.

Abgabe: 10.12.2003, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

⁷ 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)

⁸ Originalzitat: „Although a “log table” is not as useful as it once was, it still makes a good example of iteration.“

Didaktik der Informatik II -- Lösung zu Blatt 8

Aufgabe 1:

Als Beispiel ein durchgehendes Beispiel kann hier ein Login-Prozess gewählt werden. Dabei kann man auch auf die Kapselung von Objekten durch Klassen eingehen. Leider unterstützt dies Python nicht vollständig; private Methoden und Attribute gibt es nur scheinbar. So ist klar, dass es Klassen Benutzer, Benutzerliste und Einlogen (verwaltet den Prozess). Dabei werden jeweils die einzelnen Attribute gekapselt.

Die einzelnen Teile des Prozesses werden dann jeweils durch Funktionen in den jeweiligen Klassen abgebildet:

- * Einlesen des Namens des Benutzers
- * Einlesen des Passworts
- * Überprüfen des Nutzernamens/Passworts
- * ein Versuch des Einloggens
- * bei Fehlversuchen die Wiederholung von Versuchen
- * usw.

Dabei werden logischerweise auch eine Menge lokaler Variablen benötigt. Der Name des Benutzers wird hierin zum Beispiel sehr häufig verwendet. Es ist zu erkennen, dass die Variablen/Attribute lediglich in ihrem Kontext verändert bzw gelesen werden können. Um sie in einem andern Kontext sichtbar zu machen, müssen sie durch die Funktion oder eine Funktion der Klasse übergeben werden.

mögliche Aufgabestellungen zum Python Beispiel:

- * Erweitere die Klasse BenutzerListe, so dass auch Benutzer hinzugefügt werden können. Implementiere hierzu eine Methode hinzufügen.
- * Benenne die Klasse Einlogen in Benutzerverwaltung um. Und implementiere eine Methode benutzerAnmelden. Dazu müssen möglicherweise weitere Funktionen in dieser oder den anderen Klassen erstellt werden. Beachte das es keinen Nutzernamen doppelt geben darf. Die Daten für den Benutzernamen und das Passwort sollen dabei von der Tastatur gelesen werden. Wie kann man verhindern, das der Benutzer bei verdeckter Eingabe des Passworts sicher vertippt und nach der Anmeldung sein Passwort nicht mehr weiß.

Diese Beispiel lässt sich zu einer vollständigen Benutzerverwaltung ausdehnen. Jedoch verlässt man dann sehr schnell den Bereich der Kapselung und der von lokalen Variablen. Es wird dann primäre um entstehende andere algorithmische Probleme gehen.

Python Beispiel:

```
#!/usr/bin/env python

class Benutzer:
    def __init__(self,name,passwort):
        self.name=name
        self.passwort=hash(passwort)

    def gibName(self):
        return(self.name)

    def istName(self,name):
        return self.name==name

    def istPasswort(self,passwort):
        return hash(passwort) == self.passwort

class BenutzerListe:

    def __init__(self):
        self.benutzer = []
        benutzer = Benutzer("Sabine","Hallo")
        self.benutzer.append(benutzer)
        benutzer= Benutzer("Marion","1ABCf7")
        self.benutzer.append(benutzer)
        benutzer= Benutzer("Max","fisch")
        self.benutzer.append(benutzer)
        benutzer=Benutzer("Martin","schwein123")
        self.benutzer.append(benutzer)

    def istNutzerMitNamenVorhanden(self,name):
        laenge=len(self.benutzer)
        i=0
        while (i<laenge ):
            if(self.benutzer[i].istName(name)):
                return True
            i=i+1
        return False

    def gibNutzerMitNamen(self,name):
        laenge=len(self.benutzer)
        i=0
        while (i<laenge ):
            if(self.benutzer[i].istName(name)):
                return self.benutzer[i]
            i=i+1
        return None

class Einloggen:

    def __init__(self):
        self.benutzerliste= BenutzerListe()
        self.angemeldeterBenutzer=None

    def holeBenutzerName(self):
        name=raw_input("Bitte Benutzernamen eingeben: ")
        return name

    def holeBenutzer(self,name):
        benutzer=self.benutzerliste.gibNutzerMitNamen(name)
        return benutzer

    def pruefeBenutzerVorhanden(self,name):
        vorhanden=self.benutzerliste.istNutzerMitNamenVorhanden(name)
        return vorhanden

    def holePasswort(self):
```

```

import getpass
passwort=getpass.getpass("Bitte Passwort eingeben: ")
return passwort

def pruefePasswort(self,name,passwort):
    benutzer=self.holeBenutzer(name)
    if(benutzer == None):
        return False
    return(benutzer.istPasswort(passwort))

def einlogVersuch(self):
    name=self.holeBenutzerName()
    if(not self.pruefeBenutzerVorhanden(name)):
        return False
    passwort=self.holePasswort()
    if(not self.pruefePasswort(name,passwort)):
        return False
    self.angemeldeterBenutzer= self.holeBenutzer(name)
    return True

def login(self):
    if(self.angemeldeterBenutzer != None):
        print "Ein Benutzer ist bereits angemeldet"
        return

    i=0
    nichtangemeldet=True
    while((i<3) and nichtangemeldet):
        if(self.einlogVersuch()):
            nichtangemeldet=False
        else:
            i=i+1
            print("Benutzername oder Passwort falsch!")

    if(nichtangemeldet):
        print("Einloggen fehlgeschlagen. Drei falsche Versuche!")
    else:
        print("Hallo " + self.angemeldeterBenutzer.gibName() + "!")

login= Einloggen()
login.login()

```

Aufgabe 2:

Als Beispiel für Mengen von Objekten könnte man Musikstücke (wie auch Bücher, Videos usw.) wählen. Dabei hat ein Musikstück jeweils einen Interpreten, Titel, Länge, Genre als Attribut. Möglicherweise kommt noch ein Dateiname hinzu, wenn die Lieder im MP3-Format auf dem Rechner gespeichert sein sollten. Sind hier Lieder einer CD gemeint, kommt das Attribut Titelnummer hinzu.

Sortierungskriterien können hier Interpret, Titel und/oder Genre sein. (Wenn die Organisationsform eine CD ist, kann auch die Titelnummer eine Kriterium sein).

Wie schon angedeutet kann die übergeordnete Organisationsform eine CD sein. Sie kann dann ihrerseits wiederum die Attribute Interpret, Titel, Genre und evtl Länge enthalten. Da CDs nun auch wieder eine Menge von Objekten bilden: Eine CD-Sammlung.

Bei den Liedern im MP3-Format kann die Organisationsform einer MP3-Sammlung gewählt werden. Auch hier sind Unterstrukturierungen möglich, wie etwa Playlisten, die ihrerseits wiederum eine Menge von Objekten bildet.

Aufgabenstellung:

1. Modelliere für eine CD- oder MP3-Verwaltung eine Datenstruktur. Wo bei darauf zu achten ist, dass die einzelnen Lieder später nach Interpret, Titel und Genre sortiert werden können.

Bei der MP3-Verwaltung ist darauf zu achten, dass zusätzlich zu der Liste aller MP3s, die Möglichkeit gegeben sein soll, eine Auswahl von einzelnen Stücken in einer so genannten Playlist zusammenzufassen.

2. Implementiere das für Aufgabe 1 entworfene Modell. Auch dabei darauf das die Daten auch in einer Datei speicherbar und aus ihr lesbar sein sollen.

3. Teste die Implementierung aus Aufgabe 2. Achte dabei darauf, dass die Datenstruktur alle möglichen Funktionen (entfernen, löschen, hinzufügen, sortieren usw. von Objekten), die für komplette Verwaltung gebraucht werden können, vorhanden sind und richtig funktionieren.

Bei den Aufgaben kann nun in beliebige Tiefe gegangen werden. Sie lässt sich bis zu einem Projekt eine CD/MP3-Verwaltung erweitern. Wobei das Modell und die Implementierung nach obiger Form stufenweise abwechselnd entwickelt würde. Als nächstes würden die Verwaltungs-Komponenten hinzukommen.

Diese Vorgehensweise entspricht zwar nicht vollständig der objektorientierten, jedoch hat es den Vorteil das nach der jeweiligen Implementierung diese getestet wird. So fallen Fehler im Modell schneller auf und können schneller korrigiert werden. Ich denke dieses Verfahren eignet sich als Einführung in die objektorientierte Modellierung und in die Projektarbeit.



Übung zur Vorlesung „Didaktik der Informatik II“

Blatt 9

1. Aufgabe: **(3 Punkte)**

Mit der zunehmenden Vernetzung von Informatiksystemen steigt auch die Anzahl der Dienste, die von Anwendern¹ auf entfernt liegenden Informatiksystemen genutzt werden. Stellen Sie dar, wie sich die Lebenswirklichkeit der Bundesbürger im Allgemeinen und von Schülerinnen und Schülern im Speziellen durch neue „technische Steuerungs-, Informations- und Kommunikationsmedien“ (vgl. [Klafki 1991]²) in den letzten 10 Jahren verändert hat. (ca. ein bis zwei Seiten)

2. Aufgabe: **(2 Punkte)**

Welche weiteren Veränderungen erwarten Sie in den nächsten 10 Jahren durch die Benutzung von Informatiksystemen in vernetzten Strukturen?³

3. Aufgabe: **(2 Punkte)**

Ein in der Vorlesung angeführtes Beispiel entfaltet sich an der Frage: „*Warum dauert die Suche nach einer Datei auf dem lokalen Informatiksystem länger, als die Suche nach einer Phrase im WWW?*“

Zeigen Sie ausgehend von dieser Frage einen „Schnitt durch den Informatikurm“ auf, indem Sie für die Schichten „Anwendungsmethodik“, „System — Realisierung“, „Algorithmik“ und „Theorie“ jeweils kurz begründen, wie Sie sie abdecken können. Gehen Sie dabei insbesondere auf Probleme der Komplexität im informatischen Sinne ein.

4. Aufgabe: **(3 Punkte)**

Beschreiben Sie eine Einbettung der „Informatischen Komplexität“ aus der 3. Aufgabe in das Modulkonzept der Informatik. (jeweils ca. ein Absatz pro Bereich)

Abgabe: 17.12.2003 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

¹ ... und natürlich auch von Schülern

² [Klafki 1991] KLAFKI, Wolfgang: Grundzüge eines neuen Allgemeinbildungskonzepts. Im Zentrum: Epochaltypische Schlüsselprobleme. In: Neue Studien zur Bildungstheorie und Didaktik: Zeitgemäße Allgemeinbildung und kritisch-konstruktive Didaktik. Weinheim, Basel : Beltz Verlag, 1991.

³ vgl. [Maurer2001] MAURER, Hermann: Die (Informatik-)Welt in 100 Jahren, Informatik-Spektrum, S. 65-70, 2 (24), April 2001

Aufgabe 1)

Beginnend mit dem Offensichtlichen muss festgestellt werden, dass sich der Prozentsatz der deutschen (und wahrscheinlich auch weltweiten) Haushalte mit einem eigenen Computer stetig erhöht hat. Dadurch hat sich natürlich sowohl für diese Personen ganz allgemein und für SchülerInnen im Besonderen ein Tor aufgetan. Innerhalb der letzten zehn Jahre sind Dienste wie zum Beispiel die E-Mails publik geworden und die frühere Skepsis ist soweit gewichen, dass heute in vielen Unternehmen von den Bewerbungen bis zur Geschäftsführer-Korrespondenz die der Schriftverkehr weitgehend elektronisch fließt. Diese Akzeptanz spiegelt sich natürlich auch im privaten Gebrauch wieder. Die entsprechenden Programme sind einfach zu handhaben, die Verlässlichkeit der Systeme enorm und die Kosten schlichtweg minimal. Für Schüler ist es mittlerweile normal, E-Mails zu schreiben – wahrscheinlich normaler als "normale" Briefe. Nebenbei bemerkt geht durch die E-Mails und neuerdings die SMS-Nachrichten, die ja in Echtzeit verschickt werden und dementsprechend schnelle Antworten erwarten, gehörig Qualität des sprachlichen Ausdrucksvermögens der SchülerInnen verloren. Als seien diese Systeme noch nicht schnell genug, hat sich das Instant Messaging etabliert. Sofort zu wissen, wer, wo, wann, warum was macht scheint ein dadurch deckbares Bedürfnis zu sein wobei die Taktzahl und Geschwindigkeit der Nachrichten nochmals erhöht wurde.

Soviel zum Thema "Kommunikation", hat doch der bestenfalls grau-legitime Bereich des Filesharings ein noch viel größeres Bedürfnis gestillt: Wie kommt man umsonst an Musik und Filme? Nachdem der zentral Ansatz wie bei Napster eine große Schwäche offenbarte - den zentralen Server kann man sperren - gibt es nun "echte" Peer-2-Peer Tausch-Netzwerke ohne solche Server. Vor allem Schüler und Studenten nutzen diese beliebte Möglichkeit "günstig" an aktuelle Medien zu gelangen, die normalerweise, also früher, teuer hätten bezahlt werden müssen. Das alles hätte aber nichts genutzt, wäre nicht zeitgleich die Hardware zum versilbern des Beutezugs erschwinglich geworden - denn soweit waren wir vor ein paar Jahren noch nicht, dass man zum Glotzen an den PC ging oder gar zum Musik hören mit FreundInnen. Momentan ziehen die PCs ja ins Wohnzimmer ein.

Womit sich die Bedienung des PC nun endgültig zum Allgemeinwissen dazugesellen wird. Schüler sind natürlich firm darin, können den PCs bereits nach eigenen Bedürfnissen einrichten und auch programmieren. Geradezu lächerlich ist für die meisten SchülerInnen die Aufgabe einen Musiktitel am PC abzuspielen oder einen Film über die Video-Software aufzunehmen und auf CD oder DVD zu brennen. Bei der Bundesbevölkerung haben wir leider ein enormes Gefälle zwischen Kennern und Nicht-Wissern und wer nicht in nächster Zeit auf den Zug aufspringt dem droht das informatisch Abseits (Ainformatiker statt Asozial), da der Einstieg immer mehr Grundlagenwissen voraussetzt. Beispielsweise ist allgemein anerkannt, dass ein Doppelklick mit der Maus ein Objekt "öffnet" - ein paar Exoten mal ausgenommen, oder dass man sich abstrakt durch Auswahl-Menüs "bewegen" kann. Das gehört zu den informatischen Fähigkeiten, die immer mehr Schüler nicht in der Schule sondern zu Hause lernen - von Freunden oder autodidaktisch - und diese Fähigkeiten sind essentiell für die technische Steuerung von Geräten jeder Art.

Bleiben also noch die konsumierenden Dienste zu erwähnen. Dort ist ja gerade das WWW als Plattform für Informationsdienste hervorgetreten. Aber auch vermittelnde Systeme (wie beispielsweise Ebay) oder Stellenbörsen gehören dazu und werden von jedem genutzt der die Fähigkeit "surfen" besitzt. SchülerInnen haben auch hier wieder einen deutlichen Vorsprung, weil sie viel besser mit der reizvollen Informationsüberflutung zurecht kommen und Wesentliches viel besser von Werbung oder sogar irreführenden Angeboten unterscheiden können als ältere Semester - Übung macht halt immer noch den Meister. Dasselbe gilt für Suchmaschinen. Auch hier muss festgestellt werden, dass durch den regelmäßigen Umgang mit diesen Werkzeugen eine Sensibilität erreicht wurde - bei allen Personen die damit arbeiten – wie eine erfolgreiche Anfrage formuliert werden muss. Vor zehn Jahren war das recht uninteressant - da war die "Optimierung" einer Website für Suchmaschinen unbekannt, der Inhalt stellte die Werbung dar. Interessanterweise ist auch diese Sensibilität für "falsche" Informationen grundlegend, denn Personen ohne diese Fähigkeit geben möglicherweise frustriert nach dem dritten irreführenden Verweis auf, jedoch nicht ohne das Internet vorher zu verdammen.

Nur namentlich erwähnen kann man an dieser Stelle noch den Newsdienst, der immer noch lebt - und mittlerweile eine Erfahrungs-Sammlung in allen Gebieten darstellt, da hier jeder einen Schwank er

zählen kann, auch ohne eigene Homepage. In letzter Zeit scheinen sich die Dinge durch einen Schub neuer Technologien wieder zu überschlagen. Da werden Foto-Handys mit MMS-Funktion und mobilen Internet unter das Volk (unter die jugendlichen Spielkinder besonders gerne) geschwemmt, man kann mit dem Handy bezahlen, sich ausweisen und - wie es neulich eine Werbung ausdrückte - auch noch telefonieren.

Aufgabe 2)

Ohne Prophet zu sein kann man wohl davon ausgehen, dass sich die vorhandenen Strukturen weiter verkomplizieren werden aber mit einer solchen Transparenz, dass es für den Anwender einfacher scheint. Wirklich vereinfacht hat sich eigentlich noch nie etwas, weil sich etablierte Strukturen so gut wie nie ändern, es gibt immer nur einen "Aufsatz" oder bestenfalls eine neue Version oder Schnittstelle. Zum Beispiel ist es immer noch vergleichsweise umständlich, eine einfache Sat-Anlage für mehrere Endgeräte zu installieren.

Eine einfache Lösung wie beim "Kabel" - also einstecken und los - gibt es nicht und das digitale Fernsehen bringt nichts neues, sondern tauscht nur die einzelnen Komponenten aus. Großen Eifer hat ja die Klamotte mit den UMTS-Lizenzen ausgelöst, mittlerweile trauern alle um das rausgeworfene Geld und die ganzen tollen Ideen werden mit dem alten aber vorhandenen GSM-Netz umgesetzt. Nebenbei gesagt bezweifle ich, dass sich die große Mehrheit für - sagen wir - fünf Euro einen Film in Briefmarkengröße anschaut. Die immer schnelleren Computer verlieren für den normalen Anwender ihre Daseinsberechtigung, weil sie zu viel Leistung für zu viel Preis bieten. Welche Dienste auch immer dazukommen (wobei es schon fast alles zu geben scheint was man braucht), sie werden irgendwo auf dem bereits bekannten aufsetzen und damit fordern, dass man schon mal mit ein wenig Ahnung an sie herantritt.

Aufgabe 3)

- Anwendungsmethodik

Hier kann ganz einfach der Unterschied zwischen der lokalen Suche und der Internetrecherche gezeigt werden, da dieses Phänomen noch nicht jedem aufgefallen sein dürfte. Dabei kann auch gezeigt werden, dass beide Systeme die gleichen Eingaben verarbeiten müssen und gleiche Ausgaben liefern -> folglich muss es an der darunter liegenden Schicht liegen.

- System-Realisierung

Hier kann die Implementation - besonders was die Hardware betrifft - betrachtet werden. Die unterschiedlichen Ansätze der Suchmaschinenbetreiber (viele einfache PCs oder mehrere Großrechner) können dabei interessante Stichpunkte für die folgenden Ebenen liefern. Weil es für den Algorithmus eigentlich unbedeutend ist, kann hier auch besprochen werden, wie mit Suchanfragen die aus mehreren Wörtern bestehen, umgegangen wird, oder mit Anfragen bei den ein Wort ausgeschlossen wird (also ein wenig Mengenlehre bzw. Logik)

- Algorithmen

In diesen Bereich gehören schließlich die verschiedenen verwendeten Datenstrukturen und Algorithmen. Dabei kann aufgezeigt werden, welche Vor- bzw. Nachteile ein Verfahren hat. Als ein Beispiel könnte man einen statischen Index im Vergleich zum dynamischen Index betrachten und sich Gedanken über die Notwendigkeit oder die Häufigkeit der Updates machen.

- Theorie

Zuletzt können dann mathematische Verfahren zur Bestimmung der Komplexität bzw. der Effizienz (O-Notation) eingeführt werden. Da diese Verfahren erlauben von der verwendeten Hard- und Software zu abstrahieren, erlauben sie ja die relative Bestimmung der Komplexität einzelner Verfahren und bieten damit eine allgemein verbindliche Aussage zu einer Strategie oder zu einem Lösungsansatz. Dadurch werden die Schüler in die Lage versetzt, den Informatikturm sozusagen auch von unten aufzubauen.

Aufgabe 4)

- Informatiksysteme verstehen und verantwortlich nutzen

Hierbei kann die Wirtschaftlichkeit eines Informatiksystems besprochen werden im Hinblick darauf wie viel Ressourcen, die als Allgemeingut (Rechenzeit im Internet, oder Speicherplatz beispielsweise auf einem Netzwerklaufwerk) betrachtet werden müssen, durch eine bestimmte Anwendung genutzt werden. Dabei kann zunächst analytisch betrachtet werden welche Systeme wie viele Ressourcen benötigen und wie viele es minimal nur sein müssten. Nach Einführung der Komplexität könnten die SchülerInnen in Partnerarbeit Beispiele für die unterschiedlichen Komplexitäten zusammentragen und vorstellen. Es kann anhand von der Technik des virtuellen Speichers gezeigt werden, dass nicht immer ausreichend Ressourcen vorhanden sind, wodurch das Argument entkräftet wird, dass heute doch immer genügend Speicher oder Rechenleistung zur Verfügung steht.

- Modellierung

Anhand von einem Beispiel, das einen schlechten Suchalgorithmus zeigt und entsprechend lange dauert und/oder entsprechend viel Speicher benötigt, kann nach Möglichkeiten gesucht werden diesen zu verfeinern oder zu optimieren. Vergleiche mit der Suchstrategie des Menschen beispielsweise in einem Telefonbuch können dabei helfen, den SchülerInnen Anregungen zu geben. Den SchülerInnen muss der trade-off zwischen Zeit und Speicherplatz bekannt gemacht sein oder werden.

- Erkenntnisse der theoretischen Informatik

Die Kenntnisse über effiziente Datenstrukturen oder bereits bekannte Algorithmen können hier genutzt und vertieft werden, wenn die Analyse eines "langsamen" Prozesses oder eine Speicherintensiven Aufgabe betrieben wird. Dabei müssen natürlich Flaschenhälse bekannt sein - beispielsweise eine Festplattenzugriff - damit sie lokalisiert und entsprechend bewertet werden können.

Didaktik der Informatik II -- Lösung zu Blatt 9

Aufgabe 1:

Wie hat sich in den letzten 10 Jahren die Lebenswirklichkeit durch Informatiksysteme und vernetzte Strukturen verändert?

Wenn ich diese Frage in einem Satz beantworten müsste würde ich sagen: Alles und nichts! Anders gesagt: am Leben der Leute hat sich nichts besonderes geändert. Die Veränderung hat lediglich in der Art und Weise des schon vorher vorhandenen Lebens der Leute stattgefunden. -- Es haben sich lediglich die Weg des Handelns der Menschen geändert. Das Handeln selber hat es auch schon vor 10 Jahren gegeben.

In den letzten 10 Jahren haben sich Informatiksysteme stark verändert. Von der Hardware-Seite sind sie im wesentlichen schneller, billiger und kleiner geworden. Von der Software-Seite sind sie im wesentlichen bunter und damit angeblich einfacher zu bedienen geworden. Letzteres halte ich jedoch für eine Gerücht, dass von Werbeideologen verbreitet worden ist und wird.

Vor 10 Jahren ging der Boom der Heim-PCs langsam los. Heute wird fast jeder einen PC sein eigen nennen können. Das liegt zum größten Teil an der Hardware-Entwicklung. Dieser ist es zu verdanke, dass sie durch höhere Leistung viele Anwendungs-Möglichkeiten erschlossen haben. Aber auch an einer Entwicklung der Software, die mittels grafischer Benutzeroberflächen dem Nutzer vorgaukelt, es sei einfach zu bedienen.

Durch die Verbreitung von Heim-PCs gewannen auch Computernetzwerke an Bedeutung. Zunächst auf zeichenorientierten Mailboxen und später dann das Internet, das vor 10 Jahren noch unbezahlbar und wenig verbreitet war.

Es gibt fast keinen Bereich in dem noch keine Informatiksysteme Einzug gehalten haben. Für das Leben der Leute ist allerdings nur ein Hilfsmittel hinzugekommen, denn die Leute tun im Prinzip nichts anderes als vor 10 Jahren nur mit anderen Mitteln:

Photographiert wird heute mittels einer digitalen Kamera, damit können Bilder dann später leichter archiviert werden. Eine Bearbeitung von Bildern ist heute damit und mittels bezahlbaren Scannern leichter möglich, sie war es aber auch schon 10 Jahren, allerdings waren da die Mittel nicht jedermann zugänglich.

Während man früher eine Telefonzelle aufsuchen musste, wenn man unterwegs telefonieren wollte, werden heute Handies benutzt. Fast jeder ist heute fast überall auf der Welt erreichbar.

Fast alle Arten von Dokumenten werden auf Informatiksystemen gespeichert und archiviert. Während man früher seine Briefe von Hand oder mit der Schreibmaschine schrieb, werden diese heute in Computern erfasst, auch wenn Briefe nicht gleich per E-Mail versandt werden. Um an Informationen zu einem Thema zu gelangen, musste man früher in Bibliotheken nach passenden Büchern suchen und heute kann man fast alles im Internet finden.

Während man vor 10 Jahren noch von einem Geschäft ins andere laufen musste, um Dinge zu kaufen, kann man dies heute übers Internet erledigen. Filme werden heute in wesentlich besserer Qualität auf DVD geliefert. In fast jedem Elektrogerät stecken heute kleine Informatiksysteme, selbst Lichtschalter werden teilweise schon von Mikrocontrollern gesteuert. Um sich mit Leuten zu unterhalten, können Chat-Rooms im Internet verwendet werden. So kann man dann auch mit Leuten, die auf der ganzen Welt verstreut leben, Kontakte knüpfen. Musik und Videos können über das Internet bezogen werden, sowie am Computer bearbeiten oder erstellt werden. Bankgeschäfte werden heute auch schon von zu Hause aus mittels Internet erledigt.

Allerdings hat dies Entwicklung auch ihre Schatten-Seiten. Es werden von den Benutzern immer Kenntnisse der benutzen Informatiksysteme benötigt:

Firmen benutzen das Internet als ihren ganz persönlichen Spielplatz bzw. als riesengroßen Werbefläche und nerven Benutzer mit Spam-E-Mails, Werbemails und Werbe-Popups. Um dem Herr zu werden, muss der Benutzer die Informatiksysteme so weit kennen, das er geeignete Gegenmaßnahmen einleiten kann. Auch das Sammeln von Daten zu Personen in Netzstrukturen gehört zu den Problemen (Sicherheits-Behörden wird es freuen), sie können für Personen bezogene Werbung benutzt werden. Auch für kriminelle Personen ist ein Platz in solche einer vernetzten Struktur, es wimmelt nur gerade so von Viren, Würmern, Dailern und anderen unerwünschten Programmen. Fast jeder mit ein paar Computerkenntnissen kann seinen ganz privaten Rachefeldzug via Internet gegen irgendwen füh-

ren. Wobei Dialer eher in die Richtung Raub bzw. Betrug gehen. Wobei der Raub eher doch Abfangen von Bankverbindungsdaten geschieht. Niemand weiß heute so genau wie lange Medien wie CDs halten, was zu einem Problem für archivierte Daten führt. Genauso wie die der verlustbehafteten Komprimierung bei Audio- und Videodaten, ein Problem bei der Archivierung sind.

Last but not least, das Problem das Informatiksystem fast nie so funktionieren wie sie sollten. Systemabstürze und Fehler sind an der Tagesordnung, gerechtfertigt wird dies von Softwarefirmen mit der Lüge, es sei unmöglich fehlerfrei Software zu erstellen -- möglich ist dies schon, es ist nur sehr sehr aufwendig, und setzt andere Infrastrukturen der Entwicklung von Software voraus. (Ich frage mich, ob diese Leute noch ruhigen Gewissens in ein Flugzeug einsteigen, es wimmelt dort doch nur so vor fehlerhafter Software).

All diese Dinge zu beachten sind für einen Benutzer viel Arbeit. Er braucht außerdem eine Menge Wissen über Informatiksysteme. Er wird sich jedoch kaum dagegen wehren können, Informatiksysteme haben bereits Einzug in alle Bereiche des Alltags gehalten.

Aufgabe 2:

Die Entwicklung der 10 Jahre vorauszusagen liegt mir fern. Dieses hängt von vielen Faktoren ab, die mit Informatik wenig zu tun haben. Auch heute schon wird die angebliche Wissens- oder Informationsgesellschaft ausgerufen. Das es mit dem Zunehmen vernetzten Systemen wie dem Internet primäre um Wissen oder Information geht, halte ich für ein Gerücht bzw. um eine glatte Lüge. Viel mehr geht es dabei um ökonomische Interessen -- die Erschließung neuer Märkte.

Eine (gesellschaftliche) Revolution ist dort nicht zu erwarten, sie setzt politgesellschaftliche Motive voraus. Diese sind bis jetzt allerdings nur vereinzelt zu finden. Siehe dazu z.B. www.oekonux.de. Die Entwicklung der Vernetzung wird sicher weiterhin stark zunehmen. Was dann letztendlich in den Netzen angeboten wird, werden kaum noch freie Informationen sein. Damit lässt sich schließlich auch Geld verdienen.

Eingekauft wird vielleicht zukünftig nur noch via Internet. Wobei dann entsprechende Systeme, die es auch schon exemplarisch gibt, automatisiert den täglichen Bedarf bestellen. Eine weitere Perspektive ist die Telearbeit, die vermutlich zunehmen wird. Es ist für die Unternehmen billiger, die Leute zu Hause arbeiten zu lassen, als Büroräume an zu mieten. Vielleicht wird in Zukunft sogar via Internet unterrichtet.

Dieses alles hängt aber von vielen Faktoren ab: Der eine ist politischer bzw. gesellschaftlicher Natur, Wissen/Information kann gar nicht, in einem gesellschaftlichen System wie dem aktuellen, Grundlage der Gesellschaft werden. Es ist nämlich privatisiert und unterliegt absurderweise dem Eigentum eines einzelnen, der damit anstellen kann was er will, jeder andere kann auch zu dem selben Wissen gelangen (z.B. durch Forschung), allerdings gehört es geschützt durch Gesetze jemand anderem. Da mehr Wissen einen Vorsprung in der Konkurrenz im ökonomischen Sinne bedeutet, wird er dieses Wissen für sich behalten, oder nur für viel Geld verkaufen. Das ist das Gesetz der Konkurrenz bzw. der freien Marktwirtschaft, und daran ändern auch vernetzte Strukturen im Kern nichts. Sie könnten allerdings eine Grundlage dafür sein, dieses System aus zu hebeln, ob man das will muss jeder selber entscheiden.

Der andere wesentliche Aspekt, ist die Umwelt. Vermutlich wird als erstes das Trinkwasser knapp. Dies ist eine Folge der globalen Erderwärmung, die wohl schon heute niemand mehr weg diskutieren kann. Und die sich vermutlich auch in der kurzen Zeit, die dafür notwendig ist, nicht mehr aufhalten lässt. Als Folge daraus wird ein umfassender Klimawandel einsetzen. Wie der aussehen kann dafür gibt es genug Horror-Visionen. Die Aufhaltung dieser Entwicklung liegt nun wieder im gesellschaftspolitischen, eine Gesellschaft die auf Maximierung der Gewinnsteigerung basiert, kümmert sich nicht um einen Treibhauseffekt, dessen klare Auswirkungen wir vermutlich erst in 50-100 Jahren merken werden.

Was nützen uns vernetzte Strukturen, wenn wir ums Überleben kämpfen?

Aufgabe 3:

Vorbemerkung:

Die ausgehende Frage kann sicherlich auf mehrere Weisen beantwortet werden. Die Suche nach einer Datei auf einem lokalen Informatiksystem, beschränkt sich gewöhnlich auf die Suche nach einem Teil eines Dateinamens. Sucht man lokal hingegen nach einer Text-Phrase, die in einem Dokument vorkommen soll, ist dieses ungleich komplexer, als die Suche nach einem Dateinamen. Software, die letzteres vollständig und automatisiert unterstützt, ist mir leider nicht bekannt.

Aber auch die lokale Suche nach einem Dateinamen kann unterschiedlich schnell sein. Die auf Unix-Systemen übliche Suche mittels locate ist sicherlich genauso schnell wie die Suche nach einer Phrase im WWW. Allerdings kann diese Suche falsche Ergebnisse liefern, da sie auf einer Datenbank der Dateinamen basiert. Im weiteren gehe ich nun darauf ein, was für die einzelnen Schichten behandelt werden kann. Wobei die Übergänge zwischen den Schichten eher fließend zu sehen sind.

Anwendungsmethodik:

Hierin kann auf verschiedene mögliche Suchverfahren eingegangen werden. Dabei sollte hier die Anwendungsseite im Vordergrund stehen. Es sollte darauf eingegangen werden, welche verschiedenen Möglichkeiten es gibt, nach Dateien lokal zu suchen. Wobei die dabei möglicherweise unterschiedlichen Ergebnisse, beachtet werden sollten. Auch die Geschwindigkeit ist dabei von Bedeutung.

Im WWW kann auch auf unterschiedliche Verfahren eingegangen werden. Zu nennen sind hier in jedem Fall eine Suche nach Stichwörtern und eine Volltextsuche. Auch hier bei sind die unterschiedlichen Ergebnisse und Geschwindigkeiten zu beachten.

Behandelt werden kann auch, wie am besten Suchanfragen zu stellen sind. Dies ist besonders im WWW wichtig, damit man nicht hunderte von nutzlosen Treffern bei einer Suche hat.

Für die Suche im WWW können auch noch zusätzlich Sicherheitsaspekte beachtet werden. Denn gerade im Netz werden Suchergebnisse durch einzelne Anbieter verfälscht, so dass beispielweise dialer basierende kostenpflichtige Angebote schnell gefunden werden. Gemeint sind hier die gerade mal wieder in der öffentlichen Diskussion stehend Internetseiten, die scheinbar kostenlose Hilfe bei Hausaufgaben und/oder Referaten anbieten. Es werden dann jedoch automatischer Dialer installiert, die horrenden Kosten verursachen können.

System -- Realisierung:

In diesem Bereich kann auf die Systematik der unterschiedlichen Suchsystemen eingegangen werden. Dabei sollten die Modelle der verschiedenen Suchen aufgebaut werden. Im wesentlichen kann dabei zwischen Datenbank basierenden Systemen und Systemen, die den kompletten Datenbestand durchsuchen, unterschieden werden.

Es können dabei verschiedene informatische Modelle erstellt werden, beispielweise mittels objekt-orientierter Modellierung. Diese sollten den Unterschied der betrachteten Systeme klar herausstellen. An den Modellen können dann Vor- und Nachteile diskutiert werden. Das eine System benötigt z.B. mehr Speicherplatz und das andere ist dafür langsamer. Der Begriff der Komplexität kann daran schon ausführlich behandelt werden.

Algorithmik:

Es können hier unterschiedliche Suchalgorithmen behandelt werden. Dabei sollten auch Datenstrukturen beachtet werden, da einige Verfahren, wie z.B. Hashing, auf diesen basieren.

Besonderer Wert sollte dabei auf Geschwindigkeit und Speicherplatzbedarf der einzelnen Algorithmen gelegt werden.

Theorie:

Um diesen Bereich abzudecken kann man auf die Komplexität von algorithmischen Problemen, am Beispiel von Suchverfahren eingegangen werden.

Dabei ist zu beachten, dass Komplexität an sich eine Größe ist, die einem Problem anhaftet. Daraus

ergibt sich, dass es für die meisten Probleme keine optimale Lösung gibt.

Die Komplexität hat ihr Maß nämlich in Speicherplatzbedarf und Geschwindigkeit relativ zur Eingabegröße. Eine Problem schneller zu lösen hat meist zur Folge, dass man mehr Speicherplatz benötigt, und umgekehrt. Wobei dieses unabhängig vom konkreten benutzten Rechner ist.

Es können hier auch Komplexitätsklassen behandelt werden. Auch auf unlösbare Probleme in vernünftiger Zeit kann eingegangen werden. (Dabei kann das Beispiel erweitert werden, ein Optimierungsproblem ist eine Suche nach einer optimalen Lösung.)

Aufgabe 4:

Informatiksysteme verantwortlich nutzen

Hierunter fällt alles, was in Aufgabe 3 zur "Anwendungsmethodik" genannt wurde. Insbesondere gehört dazu die Methodik bei einer Suche, von der Anwenderseite aus.

Weitergehend kann darauf eingegangen werden, wie man Web-Seiten sinnvoll aufbaut, so dass Suchen schneller arbeiten und besser arbeiten, etwa durch die Vereinbarung von Stichwörtern in Kopf der jeweiligen Seite. Weiterhin soll behandelt werden, das man diese Verfahren ausnutzen kann, um in einer Suchmaschine höher aufgeführt zu werden.

Für lokale Suchen, kann man auf sinnvolle Verzeichnisstrukturen eingehen. Ein Suche die nur auf bestimmte Verzeichnisse beschränkt ist, für sicher schneller zum Ziel bzw. wird durch den Aufbau der Verzeichnisse ganz überflüssig.

Informatische Modellierung

Hierunter fällt alles, was in Aufgabe 3 zur "System -- Realisierung" genannt wurde.

Hierfür können Modelle von Suchmaschinen und der einer lokalen Suche erstellt werden. Dabei ist es wichtig, das die Datenbasis mit modelliert wird. Also die einzelnen Web-Seiten bei der Suchmaschine und die einzelnen Dateien und Verzeichnisse des lokalen Systems. Durch den Vergleich beider können Aussagen über Komplexität der Systeme gemacht werden.

Elemente der theoretischen Informatik

Für dieses Modul kann, alles was in Aufgabe 3 zu "Algorithmik" und "Theorie" genannt wurde verwendet werden, da beides zusammenhängt.

Anhand der verschiedenen Algorithmen von Suchen kann die Komplexität als solche behandelt werden. Wobei die Komplexität des Problems Suche sich nicht verändert und auch nicht verändert werden kann. Bei den Algorithmen werden nur unterschiedliche Aspekte der Komplexität zu lasten anderer verbessert. Dies ist auch schon am Modell der Web- und lokalen Suche erkennbar, und kann hier benutzt werden.



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]⁹ 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 geeignet¹⁰ –, 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

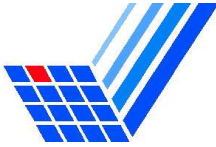
Abgabe 2. Teil: 14.1.2004, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

⁹ [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)

¹⁰ 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 II“

Blatt 12

Im Informatikunterricht sollten auch Probleme des Persönlichkeitsschutzes bzw. des Datenschutzes thematisiert werden. Eine Übersicht verschiedener Themenfelder wurde von Frau Schulzki-Haddouti in dem Buch „Bürgerrechte im Netz“ zusammengestellt (eine elektronische Quelle finden Sie im Skriptum).

Aufgabe 1:

(3 Punkte)

Treffen Sie aus den insgesamt 17 Einzelbeiträgen des Buches eine Auswahl von vier Themen, die Sie in ihrem Informatikunterricht thematisieren möchten. Begründen Sie Ihre Auswahl bezüglich der Zugänglichkeit für Kinder und Jugendliche verschiedener Altersstufen (vom Kindergarten bis zur gymnasialen Oberstufe). Berücksichtigen Sie dabei die Piagetschen Entwicklungsstufen.

Aufgabe 2:

(5 Punkte)

Skizzieren Sie eine Unterrichtsreihe (möglicherweise auch mehrere Unterrichtsreihen), in der Sie die in Aufgabe 1 ausgewählten Themen berücksichtigen. Zeigen Sie auf, wie sie die Inhalte mit weiteren Konzepten der Informatik verzahnen (Stichwort: Modulkonzept). Geben Sie in Ihrer Übersicht auch den geplanten Stundenbedarf an.

Aufgabe 3:

(2 Punkte)

Prüfen Sie an Hand des Lehrplans für die Sekundarstufe II Ihre Ergebnisse in qualitativer und quantitativer Hinsicht.

Abgabe: 21.1.2004, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

Aufgabe 1) Auswahl von vier geeigneten Themen:

a) Geistiges Eigentum im Netz

Dieses Thema halte ich für hervorragend geeignet um auch schon mit jungen Schülern - d.h. ab der Grundschule zu betrachten. Nach Piaget beginnt mit etwa 7 Jahren das Stadium der konkreten Operation. Das schließt ein, dass ein Schüler beginnt, sich in andere Menschen hineinzusetzen - geradezu ideal für Rollenspiele aller Art. Die zunehmende moralische Autonomie, die in diesem Stadium einsetzt, ermöglicht es einem Schüler darüber hinaus auch gegen die Mehrheit eine Regel abzulehnen (Beispiel: jeder lädt sich Filme aus dem Internet). Diese Voraussetzungen müssen erfüllt sein, damit Schüler überhaupt die Fragen in Bezug auf geistiges Eigentum anderer begreifen können. Ein erstrebenswertes Ziel wäre es, das moralische Rückgrat derart zu stärken, dass Schüler auch entgegen dem Gruppenzwang innerhalb des Klassenverbands agieren können. Auch in höheren Altersgruppen bleibt die Frage bestehen, wenn sich auch die Inhalte ändern (fertige Referate aus dem Netz, Software kostenlos...)

b) Anonymität im Internet

Ähnlich wie beim ersten Thema ist auch dieses geeignet um bereits in der Grundschule aufgegriffen zu werden. Mit dem von Herrn Humbert vorgestellten Planspiel zum Thema Datenschutz ("Wer hat die Oma angefahren?") ist es wahrscheinlich möglich, sogar sehr früh damit zu beginnen. Für den Kindergarten halte ich dieses Thema allerdings nicht für geeignet. Allerdings ist der Zugang für Schüler aufgrund der Relevanz im täglichen Leben sehr einfach. Nach den ersten Aha-Effekten (zum Beispiel wenn sich herausstellt, dass Anonymität im Internet oder mit dem Handy kaum möglich ist), dürfte der Grund für weitere "spannende" Fragen gelegt sein. Für ältere Schüler sollte dieses Thema sensibilisierend wirken und zwar im Hinblick auf die Preisgabe persönlicher Daten bei diversen (Zwangs-)Registrierungen, Umfragen, Chats, Payback-Punkten u.ä., zu der sie ja tagtäglich aufgefordert werden. Je nach Altersstufe und Reifegrad können verschiedene Verfahren im Unterricht vorgestellt werden um die Anonymität zu sichern oder zu untergraben.

c) Selbstdatenschutz

Ein sehr aktuelles Thema und daher auch passend für jeden Informatikunterricht. Mir fällt zwar momentan keine Methode ein um schon im Kindergarten über den Schutz der eigenen Daten zu sprechen (ich studiere hier ja auch nicht für den Kindergarten...), aber für möglich halte ich zum Beispiel eine (zugegebenermaßen sehr vereinfachte) Erklärung zur Verschlüsselung von Informationen. Möglicherweise die Übertragung von Namen in Zahlen, Farben oder Bilder. Bestimmt kann auch thematisiert werden warum manche Informationen geschützt werden müssen. ("Warum darf niemand wissen, dass ein Ersatzschlüssel unter der Fussmatte liegt?"). Zugegeben - es hinkt ein wenig, aber als Ansatz taugt es allemal. Nach Jean Piaget können ja Kindergarten-Kinder (2. Entwicklungsstufe) die Welt nur egozentrisch wahrnehmen. Ob Piaget nun Recht hat oder nicht, bleibt doch die Überlegung, dass Kinder in diesem Alter erkennen und entscheiden können was "meine Daten" und "deine Daten" sind und darauf könnte man aufbauen. In der Schule liegt das Thema quasi auf der Hand, einerseits haben die meisten Schüler vom Datenklau, Dialern und Ähnlichem gehört, andererseits wissen sie meist sehr konkret, welche Daten sie für sich gerne geschützt haben möchten. Darauf aufbauend könnte dieses Thema erörtert werden.

d) Die Zensur als technischer Defekt

Dieses Thema eignet sich am besten für eine Besprechung im Leistungskurs bzw. in der Oberstufe, da hierfür bereits ein gutes Gespür für legitime Ansprüche und fadenscheinige Begründungen zum Thema Zensur nötig sind. Vorhandenes Wissen über Zensur in der Geschichte ist notwendig um Beweggründe zu analysieren. Daher sollte hier das nötige Wissen aus Politik, Geschichte, Wirtschaft u.ä. Fachern griffbereit sein.

Aufgabe 2)

Ich würde eine Unterrichtsreihe über die Vorschläge b + c gemeinsam in etwa wie folgt durchführen:
Um deutlich zu machen wie das Modulkonzept in dieser Unterrichtsreihe zum tragen käme verwende ich abkürzend:

* - für den Aspekt: Informatiksysteme verstehen und verantwortlich nutzen

- für die informatische Modellierung

und ein ° - für die Einbettung von Erkenntnissen der theoretischen Informatik

Unterrichtsreihe:

- Welche Daten sind schützenswert, vor wem, warum. * - Eine Stunde
- Welche Daten werden offiziell gesammelt, erhoben und warum? Von der Umfrage über Cookies bis zum Zugriffsprotokoll *#° - Eine Doppelstunde
- Falsches Vertrauen: Heimtückischen Umfragen unter falschem Namen ("Wiederholen Sie ihr Ebay-Passwort"...); Adware * - Eine Stunde
- (Je nach Kenntnisstand der Schüler) Wie und wo werden Daten gespeichert? Singulär auf Wechseldatenträgern? Vielfach kopiert auf Festplatten? Zentral zugänglich im Internet? *° - Eine Doppelstunde
- Welche Daten "entstehen" beim Nutzen von Informatiksystemen (beispielsweise eigene Telefonnummern, Standort, Uhrzeit beim Versenden einer SMS)? Welche Relationen mit anderen Datenmengen sind denkbar? # - Eine Doppelstunde
- Szenario ähnlich dem Planspiel aus der Vorlesung - Eine Stunde
- Wie kann das "Entstehen" dieser Daten manipuliert oder verhindert werden? Drei Beispiele # - Eine Doppelstunde
 - a) Funktionsweise eine Proxy-Servers / Remailers
 - b) Ein "falscher" Benutzername
 - c) Eine anonyme Prepaid-Tелефонkarte
- Schutz bestehender Daten beim Sammler (Authentifizierung vor dem Zugriff, Biometrie, ACLs) #* - Eine Stunde
- Schutz bestehender Daten beim Eigentümer: Firewalls, Proxys, Verschlüsselung #* - Eine Doppelstunde
- Ausgewählte bekannte Verfahren zum Selbstdatenschutz (je nach Schülerinteresse): PGP, SSL, VPN, etc... - Drei bis vier Unterrichtsstunden
- Die Erhebung welcher Daten und Relationen kann nicht verhindert werden? - oder: Anspruch und Wirklichkeit von Anonymizern ° - Eine Stunde
- Zu welchen Daten sollte man stehen? Welche Daten sind notwendig * Eine Stunde
- Anonyme Netze #°
- Projekt (nur beim einem längeren Zeitraum für diese Unterrichtsreihe)

Da in diesem Thema sehr viele Querverbindungen existieren, wird es schwer sein, für die einzelnen Punkte genaue Zahlen für Dauer oder Umfang in Schulstunden zu schätzen. Vieles hängt ja auch vom Vorwissen und von den brennenden Fragen der Schüler ab. Ein Quartal wäre aus meiner Sicht ausreichend für diese Unterrichtsreihe wobei natürlich einige Punkte ausgelassen oder komprimiert werden können. Für Unterrichtsmethoden kommt die ganze Palette in Betracht. Für Punkte die sehr gut "live" dargestellt werden können - beispielsweise der Nutzen eines Proxy-Servers oder einer Firewall - ist natürlich eine präsentierende Form des Frontalunterrichts oder ein gemeinsames Evaluieren im Gruppenunterricht denkbar. Bei der Frage nach Beispielen dafür, welche Daten von welchen Institutionen erhoben werden, kann der Gruppenunterricht effektiv eingesetzt werden um auch empirisch (Internetrecherche) zu arbeiten. Bei allen Methoden sollte berücksichtigt werden, dass erfahrene Schüler mit der Materie bereits vertraut sein können, daher wären weiterführende Aufgaben bereitzuhalten (Welche Möglichkeiten gibt es zum Schutz einer E-Mail? Wie funktioniert PGP? Wie hoch ist die Gefahr der Kompromittierung? Wie steht es mit der Akzeptanz oder dem Komfort?) o. ä.

Aufgabe 3)

Zunächst einmal fordert der Lehrplan einen möglichst fächerübergreifenden Unterricht. Gerade beim Thema Datenschutz/Zensur sind ja die Beweggründe in den Fächern Politik und Geschichte zu suchen. Möglichkeiten zur Verschlüsselung und deren Effektivität können in Mathematik untersucht werden. Zum Inhalt fordert im Lehrplan ja gerade der Punkt 2.2.1.2 die Analyse und Bewertung von "Möglichkeiten, Grenzen, Chancen und Risiken der Informatik- und Kommunikationssysteme" mit den folgenden Teilzielen, die durch die Aufgabe 2 abgedeckt werden:

- den Einsatz von Informations- und Kommunikationssystemen in verschiedenen gesellschaftlichen Bereichen untersuchen und bewerten
- die Notwendigkeit des Verantwortungsvollen Umgangs mit Informationen einschätzen

Aufgabe 1:

Hansjürgen Garstka

Informationelle Selbstbestimmung und Datenschutz

Das Recht auf Privatsphäre

Dies ist unserer Meinung nach eins der wichtigsten Themen. Die Schüler müssen erfahren, was sie für Rechte haben, und was nicht. Auf vielen Seiten werden Angaben gefordert, die diese Rechte verletzen. Nur eine aufgeklärte Gesellschaft kann gegen so was vorgehen, und da muss ich als Lehrer den Anfang machen.

Manuel Kiper

Spione im Büro

Überwachung am Arbeitsplatz

Ein sehr wichtiger Punkt, der immer angesprochen werden sollte. Es wird sich keiner der Schüler über so was Gedanken gemacht haben. In unserem Zeitalter ist die Überwachung immer „wichtiger“?! Es geht so schnell, dass man überwacht wird und es ist meine Pflicht als Lehrer die Schüler dazu zu sensibilisieren und ihnen auch klar zu machen, das es keine Daten gibt, die nicht sensibel sind.

Ute Bernhardt

Filtern, Sperren, Zensieren?

Vom Umgang mit unliebsamen Inhalten im Internet

Dies würde ich aber erst nach der Netikette durchnehmen, weil man auf die Nettikette eingehen kann und so den verantwortungsvollen Zugang hat. Damit würden Filter eigentlich dann kaum noch notwendig sein, aber nichts desto trotz ist es wichtig, dass die Schüler die Möglichkeiten kennen lernen sollen, wie sie Inhalte filtern können.

Martin Goldmann

Dear Emily Postnews

Die Geschichte der Netikette

Dies hab ich ausgewählt, weil jeder Schüler wissen soll, was im Netz rechtens ist, und was nicht. Jeder soll sich den Strafen bewusst sein, die auf ihn zukommen, sollte er im Netz Schaden anrichten. Je mehr dies wissen und danach handeln, desto mehr andere werden mitgezogen sich auch so zu verhalten. Im wirklichen Leben, bekommt man die Regeln im Umgang mit andern Menschen ja automatisch beigebracht, im Netz ist es nicht so, das merkt man viel bei Onlinespielen. Dort ist der Ton häufig nicht sehr freundlich.

Aufgabe 2:

Beginnen würde ich mit dem Datenschutz, also dem ersten der vier Themen. Dort würde ich die Rechte ansprechen und mit den Schülern auch mal im Netz suchen, wo diese Sachen verletzt werden. An dieser Stelle ist es schwierig, auf theoretische Informatik ein zu gehen, da ich hier kein gutes Beispiel weiß. Man könnte vielleicht überlegen, wie man Daten so abspeichern kann, dass möglichst wenig Unsinn damit getrieben werden kann. Dafür würde ich 3 Stunden (mit dem theoretischen Teil 6 Stunden) einplanen.

Am Thema Datenschutz kommt man dann zum Thema des verantwortungsvollen Umgangs mit Daten und zur Netikette. Man kann sich dann gut ansehen, warum sie sich entwickelt hat, und wie wichtig sie ist. Zum theoretischen Teil lässt sich an der Frage, „warum soll man möglichst wenig Datenmüll verbreiten?“, in die Übermittlungstechnik des Internets mit seinen Protokollen etc. einsteigen. Hier würde ich 2 Stunden für die Netikette und 4-6 Stunden für die theoretischen veranschlagen.

Wenn wir dann schon beim Internet und der Netikette sind, ist es auch nicht mehr weit, die Frage zu erörtern, was man macht, wenn sich gewisse Personen dagegen widersetzen und ist dann bei Filtern usw. Hier würde ich darauf eingehen, warum gewisse Inhalte gefiltert werden müssen, z.B.: wegen dem Schutz der Kinder usw.

Dann kann die Frage erörtert werden, wie es möglich ist, diese Filter zu erstellen. So ist man schon in der theoretischen Informatik. Dafür würde ich auch so 6 Stunden angeben.

Als letztes in der Reihe würde ich dann die Spione im Büro ansprechen. Die Schüler haben dann das Wissen über Filtertechniken und Datenschutz. So könnte man auch ein Planspiel machen, wo eine Gruppe als Aufsichtsrat versucht, eine Überwachung einzuführen und begründet, warum das wichtig ist und eine andere als Arbeiter dagegen ist und durch Gesetze und Argumente dies stoppen will. Dort sollten sie dann merken, dass da vom Gesetz aus viele Lücken für die Firmen sind. 3-4 Stunden sollten dafür genügen.

Aufgabe 3:

Im Lehrplan wieder zu finden ist das Themengebiet „Datenschutz und Datensicherheit“ im Kapitel 1.2 Zusammenarbeit mit anderen Fächern (Seite 8). Hier wird das Thema mit dem gesellschaftswissenschaftlichen Aufgabenfeld des Informatikunterrichts begründet. Ebenso kann die Unterrichtsreihe am Kapitel 2.2.1.1 Modellieren und Konstruieren (Seite 12) erklärt werden. Es muss das Problem des Datenschutzes erarbeitet werden und dazu eine Lösungsstrategie für z.B. das Verhalten im Internet bestimmt werden. Auch im Kapitel 3.2 Gestaltung der Lernprozesse (Seite 37) kann das Thema wieder gefunden werden. Die Schülerinnen können ihr eigenes Wissen zum Internet und mögliches Fehlverhalten mit einbringen. Genauso sammeln sie aber auch Wissen, welches sie in der Oberstufe und genauso im späteren Studium benötigen um sich sicher im Internet zu bewegen.



Übung zur Vorlesung „Didaktik der Informatik II“

Blatt 13

Thema dieses Übungsblatts: Frauen und Informatik

Aufgabe 1:

(5 Punkte)

Auf der zweiten Seite dieses Übungsblatts finden Sie die `/etc/hosts` des Studienseminars in Hamm. Bei der Benennung der Informatiksysteme haben sich die Administratorinnen offenbar einige Gedanken gemacht. Was können Sie in einer ersten Näherung (ca. 1-2 Sätze, ggf. mit Quelle) über die Namen der Systeme sagen?

Aufgabe 2:

(5 Punkte)

Auf der Folie „Projekte mit nicht-naturwissenschaftlichen Fächern“ werden eine Reihe unterschiedlicher Themenbereiche aufgeführt.

- a) Formulieren Sie eine Projektidee, die Sie in einem (exakt!) zweiminütigen Vortrag den anderen vorstellen. Decken Sie dabei alle Bereiche dieser Folie ab. In dieser Aufgabe wird Ihre Präsentation bewertet. **(3 Punkte)**
- b) Legen Sie (schriftlich) dar, wie der Bezug zur reflexiven Koedukation durch das Thema / Inhalt / Methode hergestellt wird und wie es mit empirischen Belegen abgesichert werden könnte bzw. abgesichert werden kann. **(2 Punkte)**

Bspl.:

Roboter == Jungen?

Gestaltung == Mädchen?

Abgabe: 27.1.2004, 12⁰⁰ Uhr

Die Bearbeitungen der Aufgaben sind in elektronischer Form abzugeben.

Zweite Seite des Übungsblatts 13

	192.168.0.11	ada	ada.semsek2.ham.nw.schule.de
1+	192.168.0.88	malone	malone.semsek2.ham.nw.schule.de
2+	192.168.0.25	church	church.semsek2.ham.nw.schule.de
3+	192.168.0.26	beck	beck.semsek2.ham.nw.schule.de
4+	192.168.0.27	stec	stec.semsek2.ham.nw.schule.de
5+	192.168.0.28	bartik	bartik.semsek2.ham.nw.schule.de
6+	192.168.0.22	hopper	hopper.semsek2.ham.nw.schule.de
	192.168.0.23	babbage	babbage.semsek2.ham.nw.schule.de
7+	192.168.0.24	taylor	taylor.semsek2.ham.nw.schule.de
8+	192.168.0.2	zuse	zuse.semsek2.ham.nw.schule.de
9+	192.168.0.41	simmer	simmer.semsek2.ham.nw.schule.de
10+	192.168.0.1	turing	turing.semsek2.ham.nw.schule.de
11+	192.168.0.111	clarke	clarke.semsek2.ham.nw.schule.de
12+	192.168.0.112	peter	peter.semsek2.ham.nw.schule.de
13+	192.168.0.113	forsythe	forsythe.semsek2.ham.nw.schule.de
14+	192.168.0.114	rees	rees.semsek2.ham.nw.schule.de
15+	192.168.0.115	blanch	blanch.semsek2.ham.nw.schule.de
16+	192.168.0.116	jennings	jennings.semsek2.ham.nw.schule.de
17+	192.168.0.117	mcnulty	mcnulty.semsek2.ham.nw.schule.de
18+	192.168.0.118	bolotsky	bolotsky.semsek2.ham.nw.schule.de
19+	192.168.0.119	butler	butler.semsek2.ham.nw.schule.de
20+	192.168.0.120	gersten	gersten.semsek2.ham.nw.schule.de
21+	192.168.0.220	estrin	estrin.semsek2.ham.nw.schule.de

Didaktik der Informatik II -- Lösung zu Blatt 13

Aufgabe 1:

* Augusta Ada Byron, Countess of Lovelace lebte von 1815 bis 1852. Sie arbeitet zusammen mit Charles Babbage an seiner Analytical Engine, dem wohl ersten programmierbaren Computer, der nie gebaut wurde. Sie entwickelte Programme für diese Maschine und gilt deswegen heute als die erste Programmiererin. Ihr war schon bewusst, dass man nahezu alles durch Symbole darstellen kann und dass diese Maschine zum Manipulieren der Symbole taugt. Weiterhin machte Sie sich Gedanken, die in Richtung künstlicher Intelligenz gingen.

* Thomas W. Malone ist Professor am MIT. Er legte Grundlagen und für intelligente Systeme, die einem Benutzer helfen sollen aus einer großen Menge von Informationen, die gewünscht zu finden, zu filtern und zu sortieren. Seine Forschung bezogen sich auch auf Lernsoftware und Büroinformationssysteme. Dieses alles bereits lange bevor Computer und Netzwerke weit verbreitet waren.

* Alonzo Church war Mathematiker und beschäftigte sich mit Logik. Er erfand das Lambda-Kalkül, die Basis für jede funktionale Sprache.

* (Kent Beck hat eine alternative (?) Art und Weise der Programmierung entwickelt. Sie ist unter dem Namen Extreme Programming bekannt.)

* Milly Beck, Gail Taylor, Patsy Simmers und Norma Stec werden häufig als die ersten vier Bezeichnet. Sie hatten irgendetwas mit dem ENIAC, einem der ersten universal Rechner, zu tun. Ich nehme an, dass sie auch zu den Programmierern gehörten.

* Grace Murray Hopper hat den ersten Compiler und später Cobol entwickelt.

* Charles Babbage lebte von 1791 bis 1871. Er entwickelte detaillierte Pläne für Rechenmaschinen zusammen mit Ada Byron. Darunter war auch die Analytical Engine, dem wohl ersten programmierbaren Computer. Ihm selbst waren die Möglichkeiten, die diese Maschine bot, nicht bewusst. Ada Byron machte sich mehr Gedanken in Richtung der universellen Programmierbarkeit der Maschine. Sie wurde jedoch vermutlich nie gebaut.

* Konrad Zuse war der Entwickler des ersten voll funktionsfähigen und programmierbaren digitalen Computers. Für diesen entwickelte er das Plankalkül.

* Alan Turing war Mathematiker. Er erfand die Turing Maschine und beschäftigte sich mit künstlicher Intelligenz. Im übrigen gehörte einer Gruppe an, die bis heute diskriminiert wird und zu seinen Lebzeiten verfolgt wurde: Er war schwul, und wurde dafür auch 1952 verurteilt.

* Peter Chen entwickelte das Entity-Relationship Modell.

* Kay McNulty und Jean Jennings Bartik zwei der Programmierern des ENIAC, einem der ersten universal Computer.

Aufgabe 2:

a)

Hier kann zum Beispiel eine Projekt zur Automatisierung gewählt werden. Die Gestaltung eines Automaten gehört in den ersten Bereich. Auswirkungen und Gründe der Automatisierung gehören in den dritten und zweiten Bereich.

b)

Mir ist zwar nicht klar was reflexive Koeduktion sein soll, gehe hier aber von der in den Beispielen dargestellten Rollen Zuordnung aus.

Die Zuordnung von einem Geschlecht zu einem Thema wird wie folgt vollzogen: Jungen sind eher technisch/handwerklich interessiert, Mädchen eher künstlerisch/literarisch.

Eine empirische Absicherung kann sicherlich über Kurs- bzw. Berufswahlen stattfinden. Hierin dürfte erkennbar sein, dass sich die Menschen der Zuweisung entsprechend verhalten.

Um eine Absicherung bezüglich des gesellschaftlich durchgesetzten Denkens zu erhalten reicht allgemeine Erfahrung aus; man kann das in vielen Gesprächen erkennen (beispielweise in Kneipen und dergleichen). Wobei die Frage ob es so ist, ziemlich uninteressant ist. Die Gründe für der Zuordnung entsprechendem Verhalten und der allgemeinen Anerkennung dieser Zuordnung sind zu betrachten. Hierzu kann Empirie nicht taugen, da sie lediglich einen statistischen Zustand oder eine Änderung des selben beschreibt.