

Studienseminar für Lehrämter an Schulen Hamm
Seminar für das Lehramt an Gymnasien und Gesamtschulen
Stadthausstr. 3
59065 Hamm

Hamm, 31. Oktober 2016

UNTERRICHTSENTWURF

(Unterrichtsbesuch in Informatik)

**Einführung des *Klassen*begriffs durch Gruppierung ähnlicher Objekte und
Zusammenfassung gemeinsamer Attribute und Methoden.**

Referendar: - (Referendar)
Lerngruppe: Informatik GK 11
(18 Schülerinnen und Schüler, 5 weiblich und 13 männlich)
Datum: - (Datum)
Zeit: 13.25 Uhr – 14.10 Uhr (7. Stunde)
Ausbildungsschule: Gymnasium
Raum: Informatikraum

Ausbilder und Schulvertreter

Ausbildungslehrer: bdU
Ausbildungskordinator: -
Schulleiter: -
Hauptseminarleiter: -
Fachleiter Informatik: Dr. L. Humbert
Fachleiterin 2. Fach: -

1 Thematischer Zusammenhang

Die fachlichen Inhalte des Schulfachs Informatik unterteilen sich gemäß der Richtlinien und Lehrpläne für das Fach Informatik in der Sekundarstufe II in die zwei Kategorien „Modellieren und Konstruieren“ sowie „Analysieren und Bewerten“ (vgl. [MSWWF-NW 1999, S. 10]). Dazu gehört unter anderem der Punkt „Ein Informatikmodell gewinnen: Probleme eingrenzen und spezifizieren, reduzierte Systeme definieren“ (ebenda).

Der Anfangsunterricht in der Jahrgangsstufe 11 beginnt gemäß des schulinternen Curriculums des Gymnasiums mit dem Themenkomplex „Einführung in die objektorientierte Programmierung“¹ (OOP).

Darüber, dass der Anfangsunterricht sinnvollerweise mit dem Inhalt „Objektorientierte Modellierung“ (OOM) beginnen sollte, welcher zur OOP hinführt, konnte unter den Referendarinnen und Referendaren des Fachseminars Informatik am Seminar Gymnasium/Gesamtschule des Studienseminars für Lehrämter an Schulen Hamm schnell Einigkeit erzielt werden.

Die Teilnehmerinnen und Teilnehmer des Fachseminars entwickelten verschiedene Szenarien, die als einfache oder bereits relativ komplexe Modellierungsübungen dienen können. Dabei soll stets von Objekten ausgegangen werden, bevor später der Klassenbegriff eingeführt wird. Durch den vorgesehenen zeitlichen Abstand soll eine Vermengung der Begrifflichkeiten „Klasse“ und „Objekt“ – eine der häufigsten Fehlerquellen auch noch bei Informatikstudenten – weitgehend ausgeschlossen werden.

1.1 Gegenstand der Unterrichtsreihe

Zur Vorbereitung auf eine spätere objektorientierte Implementierung erfolgt zunächst die Modellierung eines einfachen Sachverhalts: Bei einem Basketballturnier treten vier Mannschaften gegeneinander an (Spielmodus: Liga). Nachdem alle Mannschaften ihre Spiele absolviert haben, soll der Gesamtsieger ermittelt werden.

Da nur eine Minderheit des Kurses (vier männliche Schüler) einen Differenzierungskurs Informatik der Jahrgangsstufen 9/10 besucht haben und nach eigenen Angaben über Vorkenntnisse im Bereich der objektorientierten Programmierung verfügen, wurde dieses einfache Problem einem möglicherweise interessanteren (aber auch komplizierterem) vorgezogen.

Anhand des Basketball-Problems soll der Vorgang der Softwareentwicklung – ausgehend von einer Problembeschreibung über eine informatische Modellierung bis hin zur Implementierung – nachvollzogen werden.

1.2 Thema der Unterrichtsstunde

Einführung des *Klassen*begriffs durch Gruppierung ähnlicher Objekte und Zusammenfassung gemeinsamer Attribute und Methoden.

¹Der Begriff der objektorientierten Programmierung kann an dieser Stelle nicht erläutert werden. Stattdessen sei auf den Eintrag „objektorientierte Programmierung“ in [Claus und Schwill 2006, S. 466ff] verwiesen.



1.3 Struktur der Unterrichtsreihe

Nach einer Eingangsbefragung der Schülerinnen und Schüler mittels eines anonymen Fragebogens (hinsichtlich ihrer Vorkenntnisse, Erwartungen und Wünsche bezüglich des Informatikunterrichts sowie einer Aufforderung, den Begriff Informatik für sich selbst zu definieren), wurden die Teildisziplinen der Informatik kurz beleuchtet und den Schülervorstellungen zugeordnet.

Der Einstieg in die OOM erfolgte durch Anwendung des Verfahrens nach ABBOTT für eine gegebene Problembeschreibung (Basketballturnier, siehe Anlagen). Es wurden mögliche Objekte und deren Attribute und Methoden identifiziert und in Objektkarten eingetragen. Dabei ergaben sich mitunter kontroverse und fruchtbare Diskussionen.

Weitere Unterrichtsgespräche wurden über die Beziehungen zwischen den Objekten und die Interobjekt-Kommunikation geführt (kombinierte Info-/Arbeitsblätter zu Objekt- und Sequenzdiagrammen finden sich im Anhang). Die Schülerinnen und Schüler entwickelten weitgehend selbständig ein informatisches Modell des Basketballturniers und erstellten dazu ein Objektdiagramm, welches auf Tapetenrollen festgehalten wurde.

1.4 Lernziele der Unterrichtsstunde

Hauptlernziel: Die Schülerinnen und Schüler sollen formulieren können, was eine Klasse ist und die Unterschiede zwischen Klassen und Objekten herausstellen können.

Feinziele:

- Die Schülerinnen und Schüler abstrahieren ähnliche Objekte, indem sie die Gemeinsamkeiten formulieren und Oberbegriffe finden.
- Die Schülerinnen und Schüler leiten aus Objekten eines Objektdiagramms die Klassen eines Klassendiagramms ab.
- Die Schülerinnen und Schüler konzipieren Klassen ohne den Umweg über Objekte.

2 Hausaufgaben zur Stunde

Die bisher erstellte Modellierung ist weitgehend tragfähig und kann vorerst als abgeschlossen gelten.² Aus diesem Grund wurde in der vorangegangenen Stunde keine nachbereitende Hausaufgabe gestellt.

Der Übergang von Objekten zu Klassen ist ein zwingender Schritt in der eingangs erwähnten didaktischen Konzeption (vgl. Abschnitt 1) und somit eine konsequente Fortsetzung des bisherigen Unterrichts.

²Einige Mängel in Hinblick auf eine effiziente und „wartungsarme“ Implementierung (z.B. beidseitige Kennt-Beziehungen zwischen den Spielen und den Mannschaften) sind noch vorhanden, sollen jedoch erst in der Implementierungsphase aufgegriffen werden, da die Problematik zum jetzigen Zeitpunkt den Schülerinnen und Schülern nur schwer vermittelbar wäre.



Um dennoch eine kleine Zäsur zum bisherigen Unterrichtsverlauf zu setzen (es soll deutlich herausgestellt werden, dass die Modellierung eine neue Ebene erreicht hat), wurde auch auf eine vorbereitende Hausaufgabe verzichtet. Folglich kann die zur Verfügung stehende Unterrichtszeit vollständig für die Einführung des Klassenbegriffs genutzt werden.

3 Geplanter Unterrichtsverlauf

Unterrichtsphasen	Operationen/Sachaspekte	Aktions- und Sozialformen	Medien
Stundenbeginn	Begrüßung und Vorstellung der Gäste	LV	–
Einstieg & Problemstellung	Betrachtung des abgeschlossenen Objektdiagramms; bedeutet die hohe Anzahl an Objekten einen großen Programmieraufwand oder gibt es Gemeinsamkeiten, die sich ausnutzen lassen?	UG	Objektdiagramm an der Fachraumwand
Erarbeitung I	Sammeln von Gemeinsamkeiten, bilden von Kategorien (Klassen) von Objekten	UG	s. o.
(Vertiefung I)	Sofern dies im Unterrichtsgespräch noch nicht deutlich wurde, sollen weitere Abgrenzungen des Klassenbegriffs zum Objektbegriff herausgestellt werden	LV	–
Vertiefung (II)	Erläuterungen zur grafischen Darstellung von Klassen (Klassendiagramm)	EA	AB 4
Erarbeitung II	Die SuS wenden ihre neu erworbenen Kenntnisse über Klassen an (indem sie unter Zuhilfenahme des von Ihnen zuvor angefertigten Objektdiagramms ein Klassendiagramm erstellen) und präsentieren ihre Lösung	GA	AB 4, OHP-Folien (teilweise als Schablone)
Festigung	Die SuS üben das direkte Modellieren einer Klasse anhand eines einfachen Beispiels (z. B. Automobil)	UG	Tafel

Der Zeitbedarf für die Bearbeitung der Gruppenarbeit lässt sich im Voraus schlecht abschätzen, da insbesondere der Aufwand zur Besprechung der Schülerlösungen schlecht vorhersagbar ist. Falls alle Schülerinnen und Schüler zu einer ähnlichen Lösung gelangen, sollte die Vorstellung einer Lösung und die Diskussion der Unterschiede zur Lösung anderer Gruppen schnell vonstatten gehen. Andererseits könnten extrem verschiedene Lösungen eine längere Nachbesprechung erfordern. Deshalb ist die Festigungsphase in dieser Stunde als optional anzusehen. Mit ihr könnte die Folgestunde beginnen.



Umgekehrt kann die geplante Hausaufgabe (siehe unten) vorgezogen werden, falls der unwahrscheinliche Fall eintritt, dass die beschriebenen Phasen weniger Unterrichtszeit erfordern als vorgesehen.

4 Geplante Hausaufgaben

Als vorbereitende Hausaufgabe sollen sich die Schülerinnen und Schüler mit der Definition einer Klasse in der Programmiersprache Python auseinandersetzen. Per Analogieschluss ist die Klassendefinition einer einfachen Klasse (Mannschaft) aus einer vorgegebenen Klassendefinition einer weiteren einfachen Klasse (Sporthalle) zu erstellen, siehe Arbeitsblatt 5.

5 Lehrsituation und didaktisch-methodische Begründungen

5.1 Lehrsituation

Der Kurs der Jahrgangsstufe 11 ist sehr angenehm zu unterrichten. Meiner subjektiven Theorie zufolge liegt dies daran, dass sich die Schülerinnen und Schüler nun als Oberstufenschüler begreifen. Sie können zumeist auch in der 7. Stunde noch konzentriert arbeiten, auch Unterrichtsstörungen sind selten.

Im Vergleich zum Informatikraum 1 ist die Akustik wesentlich besser und die Ausstattung mit Informatiksystemen moderner. Letztere wurden bislang jedoch kaum eingesetzt – eine Ausnahme bildete eine Webrecherche zur Definition des Informatikbegriffs. Dies führte vereinzelt zu Unmutäußerungen, da einige Schüler lieber am Rechner programmieren würden, wie dies im Parallelkurs der Fall ist. Mit Verweis auf moderne fachdidaktische Konzepte und die gängige Modellierungspraxis in der Softwareentwicklung wurde diesen Schülerwünschen jedoch nicht nachgegeben. Es wurde jedoch erneut darauf hingewiesen, dass die Reihenplanung den Beginn der Implementierungsphase nach dem baldigen Abschluss der Modellierung vorsieht (Transparenz der Planungsprozesse).

5.2 Inhaltliche Begründungen

Die Relevanz der informatischen Modellierung für den Informatikunterricht ist unstrittig, zumal die Informatik von vielen Informatikern als „Modellierungswissenschaft“ verstanden wird. Dementsprechend hat der Inhalt „Modellieren und Konstruieren“ auch einen großen Stellenwert in den Lehrplänen erhalten (vgl. [MSWWF-NW 1999]).

Die Fachwissenschaft hat mehrere Standards für Modellierungssprachen (zumeist in Diagrammform) hervorgebracht, um Probleme der realen Welt als informatische Modelle abbilden zu können. Als bekannteste Sprache ist hier die UML (Unified Modelling Language, siehe [Dumke 2007] für eine kurze Einführung) zu nennen, welche allerdings so komplex ist, dass sie im allgemeinbildenen Informatikunterricht auf ihre wesentlichen Konzepte reduziert werden muss.

Dabei eröffnen sich einige didaktische Gestaltungsmöglichkeiten, die, wie bereits in Abschnitt 1 erwähnt, im Fachseminar diskutiert wurden. Der verfolgte Ansatz geht vom Konkreten – den Objekten – aus und abstrahiert diese zu Kategorien von Objekten, den Klassen.



Die Klassen wiederum stellen das Bindeglied zu einer programmiersprachlichen Umsetzung dar, die demnächst durch den Kurs übernommen werden soll. Weil alle Objekte, die eine Instanz derselben Klasse sind, über vergleichbare Eigenschaften und Funktionalitäten verfügen, ist es logisch und zweckmäßig, sie innerhalb einer Klasse zusammenzufassen. Die einzelnen Objekte einer Klasse unterscheiden sich lediglich noch in den Ausprägungen ihrer Eigenschaften.

5.3 Methodische Begründungen

Bei der Konzeption der Stunde wurde auf einen ausgewogenen Anteil von Schüler- und Lehreraktivität und eine abwechslungsreiche und klare Phasierung des Unterrichts geachtet, was gerade in einer siebten Unterrichtsstunde kurz vor dem Wochenende ein wichtiger Aspekt ist.

In der Gruppenarbeitsphase haben die Schülerinnen und Schüler die Möglichkeit, untereinander über ihre Vorstellungen zur Umsetzung des Arbeitsauftrags zu diskutieren, sodass eine womöglich vorhandene Hemmschwelle bei der Anwendung des neuen Wissens niedriger anzusetzen ist als in einer Plenumsphase.

Eine programmiersprachliche Umsetzung der Klassen an den Informatiksystemen ist nicht vorgesehen, da die Schülerinnen und Schüler bisher fast nicht mit ihnen gearbeitet haben und eine Einführung in den Aufruf des Python-Interpreters und des zu verwendenden Editors sowohl zeitlich als auch konzeptionell nicht integrierbar ist. Dementsprechend ist auch die vorbereitende Hausaufgabe „theoretischer Natur“.

6 Geplantes Tafelbild

Die Tafel (bzw. das „Whiteboard“) soll in dieser Stunde lediglich in der Festigungsphase zum Sammeln der Attribute und Methoden der einfachen Beispielklasse dienen, für welche direkt eine Klassenkarte erstellt werden soll.

Klassendiagramme als Ergebnisse der Gruppenarbeiten werden auf Folien erstellt, um per Overhead-Projektor präsentiert werden zu können.

7 Übersicht der Anlagen

Bisher herausgegebene Info-/Arbeitsblätter zur objektorientierten Modellierung:

- „Das Verfahren nach Abbott; Objektkarten erstellen“ (Infoblatt 1)
- „Problembeschreibung: Basketballturnier“ (Arbeitsblatt 1)
- „Darstellung der Beziehungen zwischen Objekten in Objektdiagrammen“ (Arbeitsblatt 2)
- „Interaktion von Objekten: Sequenzdiagramme“ (Arbeitsblatt 3)
- „Beschreibung der Methoden eines Objekts“ (Infoblatt 2)
- „Spielplan des Basketballturniers“ (Ergänzung der Problembeschreibung, Infoblatt 3)



Info-/Arbeitsblätter für diese Stunde:

- Darstellung des von den Schülerinnen und Schüler erstellten Objektdiagramms (Infoblatt 4)
- Informations- und Arbeitsblatt zur grafischen Darstellung von Klassen (Arbeitsblatt 4)
- Informations-/Arbeitsblatt zur Klassendefinition in Python (Arbeitsblatt 5)

Literatur

- [Claus und Schwill 2006] CLAUS, Volker ; SCHWILL, Andreas: *Duden Informatik A-Z. Fachlexikon für Studium, Ausbildung und Beruf*. 4. Auflage. Mannheim : Dudenverlag, 2006. – ISBN 3-411-91016-X
- [Dumke 2007] DUMKE, Reiner R.: *UML-Tutorial*. <http://python.org/doc/current/tut/tut.html>. 2007. – zuletzt geprüft: 06. September 2007
- [MSWWF-NW 1999] MINISTERIUM FÜR SCHULE UND WEITERBILDUNG, WISSENSCHAFT UND FORSCHUNG DES LANDES NORDRHEIN-WESTFALEN (Hrsg.): *Richtlinien und Lehrpläne für die Sekundarstufe II – Gymnasium/Gesamtschule in Nordrhein-Westfalen – Informatik*. 1. Aufl. Frechen : Ritterbach Verlag, Juni 1999 (Schriftenreihe Schule in NRW 4725)
- [van Rossum 2006] ROSSUM, Guido van: *Python Tutorial*. <http://python.org/doc/current/tut/tut.html>. September 2006. – zuletzt geprüft: 01. September 2007

