

Schriftliche Abiturprüfung Informatik

Beispiele für
schriftliche Prüfungsaufgaben 2008



Freie und Hansestadt Hamburg
Behörde für Bildung und Sport

Impressum

Herausgeber:

Freie und Hansestadt Hamburg
Behörde für Bildung und Sport
Amt für Bildung – B 21 –
Hamburger Straße 31, 22083 Hamburg
Alle Rechte vorbehalten

Referat Mathematisch-naturwissenschaftlich-technischer Unterricht

Referatsleitung: Werner Renz

Fachreferentin: Monika Seiffert

Redaktion:

Tammo Ricklefs
Jan Schöttler
Dirk Schütt
Christian Siegel
Jörg Viole

Internet: ?

Hamburg 2007

1 Was ist neu?

Grundlage der vorliegenden Handreichung ist die Richtlinie für die Aufgabenstellung und Bewertung der Leistungen in der Abiturprüfung 2007. Sie gilt erstmals für das Abitur 2008.

Im Folgenden wird auf Änderungen gegenüber der alten Abiturrichtlinie und auf zusätzlich auf Vorgaben hingewiesen, die in der Vergangenheit nicht immer im Blick der Aufgabensteller standen. **Die Lektüre dieser Darstellung ersetzt jedoch nicht die Lektüre der neuen Abiturrichtlinie!**

Im Gegensatz zu den bisherigen Vorgaben für die schriftliche Abiturprüfung im Fach Informatik bearbeiten die Prüflinge nicht mehr drei, sondern nur noch **zwei Aufgaben**. Wie bisher können die Prüflinge während der Abiturklausur nicht auswählen, welche Aufgaben sie bearbeiten wollen.

Dem Amt für Bildung sind drei voneinander unabhängige, etwa gleichgewichtige Aufgaben einzureichen, die sich auf unterschiedliche Kurshalbjahre beziehen. Das Amt für Bildung wählt die zwei zu bearbeitenden Aufgaben aus.

Im Grundkurs beträgt die Bearbeitungszeit für die Prüflinge **vier**, im Leistungskurs **fünf Zeitstunden**. Der eigentlichen Bearbeitungszeit geht eine halbstündige Vorbereitungszeit voraus, in der die Aufgaben gelesen und letzte Fragen beantwortet werden können.

Jede Aufgabe behandelt ein anwendungsbezogenes **Gesamtpproblem** einschließlich einer Reflexion hinsichtlich der Modellierung oder der Möglichkeiten und Grenzen informatischer Verfahren. Jede Aufgabe kann in Teilaufgaben gegliedert sein, die einen inneren Zusammenhang aufweisen, sich aber dennoch möglichst unabhängig voneinander bearbeiten lassen. Die Anzahl der Teilaufgaben sollte klein sein. Die Teilaufgaben sollen nicht zu kleinschrittig die Lösung des Gesamtproblems vorstrukturieren. Sie sollen hinreichend offen und möglichst auf verschiedenen Wegen lösbar sein.

Die Aufgaben sollten mehr Denk- als Programmieraufgaben sein, informatische Sachargumentation verlangen und auch grafische Darstellungen einbeziehen.

Bei der Formulierung der Aufgaben sind die Operatoren zu verwenden, die in der Richtlinie aufgelistet sind. Damit werden gezielt bestimmte Kompetenzen der Schülerinnen und Schüler angesprochen.

Die Anforderungen der Aufgaben müssen sich auf mindestens zwei der folgenden vier Themenbereiche des Rahmenplanes beziehen:

- Grafiksysteme
- Kommunikation
- Möglichkeiten und Grenzen maschineller Intelligenz
- Simulation dynamischer Systeme oder Robotersysteme

Nicht zugelassen sind:

- ausschließlich aufsatzartig zu bearbeitende Aufgaben,
- Aufgaben, die eine überwiegend mathematische Bearbeitung erfordern,
- Aufgaben ohne Kontextorientierung,
- übernommene Aufgaben (z. B. von Verlagen) ohne Zuschnitt auf den Kurs bzw. ohne Berücksichtigung der spezifischen unterrichtlichen Voraussetzungen.

Zu jeder Aufgabe sind die für die Prüflinge zugelassenen **Hilfsmittel** anzugeben.

Computer sollen während der schriftlichen Abiturprüfung möglichst nicht verwendet werden. Der Einsatz eines Computers ist nur dann sinnvoll, wenn eine Klassenbibliothek eingesehen oder Ergebnisse mit dem Computer erzeugt und anschließend interpretiert oder reflektiert werden sollen. Es muss sichergestellt sein, dass zur Prüfung genügend funktionsfähige Computer verfügbar sind und durch ihre Benutzung keine Kommunikationsmöglichkeit für die Prüflinge entsteht.

Jeder Aufgabe wird ein Vorsatzblatt zugeordnet, das entsprechend ausgefüllt werden muss. (s. Anlage Vorsatzblatt). Die Angaben zu den unterrichtlichen Voraussetzungen sowie die Erwartungshorizonte sollten nicht auf dem Vorsatzblatt untergebracht werden, da der dort verfügbare Platz unzureichend ist. Besser ist es, diese Angaben als Anlage beizufügen und unten auf dem Vorsatzblatt auf die Anlage zu verweisen.

2 Erwartungshorizont und Anforderungsbereiche

Die Abiturrichtlinie fordert, dass bei der Prüfungsaufgabe das Schwergewicht der zu erbringenden Prüfungsleistungen im Anforderungsbereich II liegt und daneben die Anforderungsbereiche I und III berücksichtigt werden, und zwar Anforderungsbereich I in höherem Maße als Anforderungsbereich III. Kurz zusammengefasst:

Anteil II > Anteil I > Anteil III

Die im Abschnitt 3.5.3 der Abiturrichtlinie festgelegten Benotungsmaßstäbe sollten bereits bei der Erstellung der Aufgaben und der zugehörigen Erwartungshorizonte mit bedacht werden. Da für die Erlangung der Note „gut“ mindestens 75 % der erwarteten Leistungen erbracht werden müssen, darunter neben Leistungen in den Anforderungsbereichen I und II auch Leistungen im Anforderungsbereich III, ist es sinnvoll, den Anteil für den Anforderungsbereich III größer als 25 % anzusetzen.

Beispielsweise könnte die Aufteilung der erwarteten Leistungen auf die Anforderungsbereiche etwa 33% für den Anforderungsbereich I, 40 % für den Anforderungsbereich II und 27 % für den Anforderungsbereich III vorsehen.

Für jede Aufgabe wird die Gewichtung der einzelnen Teilaufgaben durch die Angabe von Bewertungseinheiten in Prozent deutlich gemacht. Die Bewertungseinheiten werden den drei Anforderungsbereichen zugeordnet. Dazu sollte wie bei den folgenden Beispielaufgaben eine tabellarische Darstellung gewählt werden. Für den Prüfer der Aufgaben einerseits und den Zweitkorrektor andererseits stellt diese Darstellung eine gute Hilfestellung und Informationsquelle dar.

Die Erwartungshorizonte müssen so ausführlich dargestellt werden, dass der Zweitkorrektor hinreichend detailliert informiert wird, was der Aufgabensteller von den Prüflingen erwartet hat. Das bedeutet, dass die Lösungen konkret ausformuliert werden müssen.

3 Beispielaufgaben

Die im Folgenden vorgestellten Beispielaufgaben beziehen sich auf die Themenbereiche der ersten drei Semester der Studienstufe.

- Objektorientierte Modellierung von Grafiksystemen
- Kommunikation / Kryptologie
- Möglichkeiten und Grenzen maschineller Intelligenz

Grundkursaufgaben sind mit GK, Leistungskursaufgaben mit LK gekennzeichnet.

3.1 Hausbau

GK

Es soll ein einfaches CAD-Programm entwickelt werden, das bei der Planung des Baus von Ein- und Zweifamilienhäusern aus Fertigbauelementen hilft. Dieses Programm soll Käufern zur Verfügung gestellt werden, damit sie sich ihr Traumhaus planen können. Es soll erst einmal nur der Grundriss gezeichnet werden können. Dabei müssen die Bauherren jeweils aus einem bestehenden Katalog vorgegebener Elemente auswählen. In dem Katalog sind die Fertigbauelemente enthalten. Es gibt beispielsweise mehrere Außen- und Innenwände, Türen, Fenster und Treppen. Da die Wandelemente auch transportiert werden müssen, gibt es sie in 5 Längen zwischen 2 m und 4 m. Längere Wände müssen aus Teilwänden zusammengesetzt werden.

Die Skizzen in der Anlage zeigen den Grundriss einer zweigeschossigen Wohnung. Es soll mit dieser Skizze nur gezeigt werden, wie beispielsweise Treppen, Türen, Wände und Fenster dargestellt werden.

- Beschreiben Sie typische Interaktionen des Anwenders mit dem Programm. Entwickeln Sie daraus grundlegende Anforderungen an das Programm.
- Geben Sie Kriterien an, wann der Einsatz von Vererbung bei der objektorientierten Modellierung angemessen ist. Beschreiben Sie eine mögliche Alternative zur Vererbung.
- Entwickeln Sie für dieses System ein Klassendiagramm. Geben Sie wesentliche Attribute und Methoden an. Begründen Sie ausführlich Ihre Entscheidungen bei der Modellierung.
- Alle Elemente eines Geschosses müssen gemeinsam verwaltet werden. In Java kann dies mit Hilfe einer ArrayList implementiert werden. Beschreiben Sie das Konzept der ArrayList und die Verwendung in diesem Fall.
- Gehen Sie jetzt davon aus, dass alle Elemente eines Geschosses mit Hilfe einer ArrayList verwaltet werden und jedes Element über eine Methode gibPreis() verfügt. Implementieren Sie eine Methode gibGesamtpreis und erläutern Sie Ihre Implementation.

Kommentar

Es ist sinnvoll, die Bewertung des Anwendungskontextes als weiteren Aufgabenteil zu ergänzen. Dazu müsste ein aktueller „authentischer“ Text vorgegeben werden (z.B. von der Architektenkammer über Selbstplanung von Häusern), zu dem die Prüflinge Stellung nehmen müssten.

Unterrichtliche Voraussetzungen

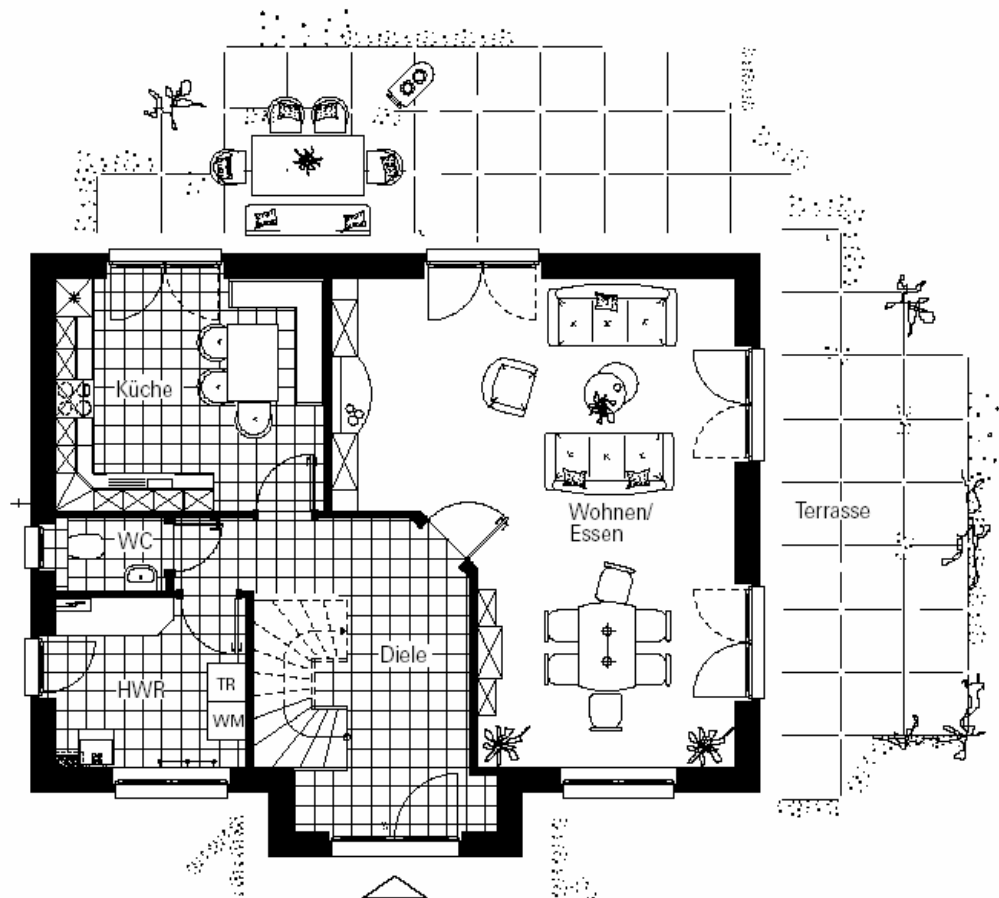
Die Aufgabe setzt voraus, dass sich die Prüflinge die im Rahmenplan genannten Konzepte der objektorientierten Modellierung und Programmierung erarbeitet haben.

Dazu gehört u. a. die Erstellung von Use-Case- und Klassendiagrammen (UML), die Eigenschaften verschiedenartiger Beziehungen zwischen Klassen sowie die Konzepte der Kapselung, der Vererbung und der Polymorphie. Ein Schwerpunkt stellte die Beurteilung wichtiger Modellierungsalternativen dar, dazu gehört beispielsweise die Alternative Vererbung / Delegation. Modelle sind mit Hilfe Java implementiert und Klassenbibliotheken genutzt worden. In Java wurden die einfachen Datentypen, der strukturierte Datentyp Array und einfache Konzepte der Collections behandelt.

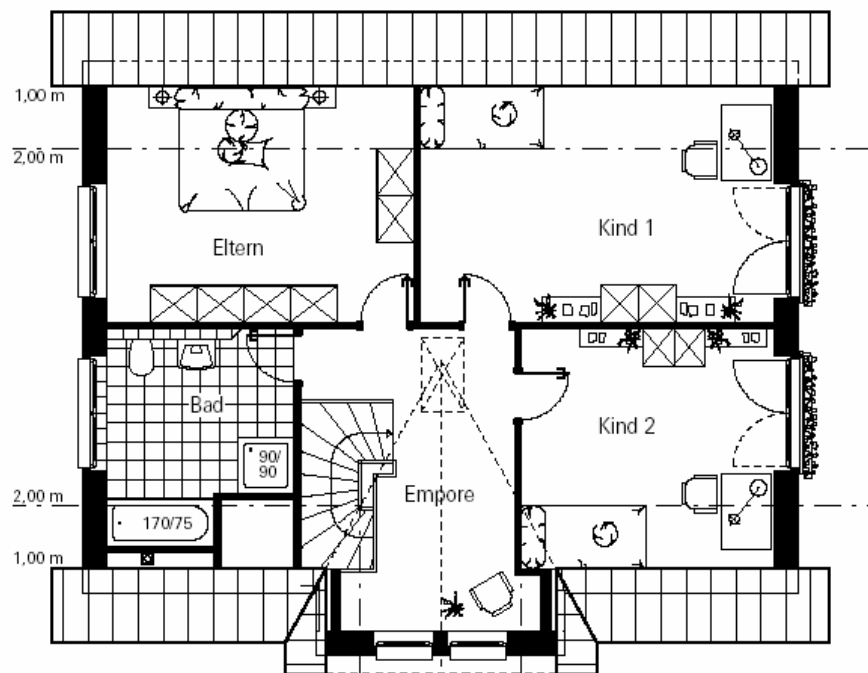
Hilfsmittel: Dokumentationen von Klassenbibliotheken

Anlage zur Aufgabe Hausbau

Erdgeschoss



Dachgeschoss



Erwartungshorizont

	Lösungsskizze	I	II	III
a)	Der Anwender erstellt Geschosse, wählt, positioniert, verschiebt, dreht, löscht Bauelemente, er lässt den Plan zeichnen, er lässt den Preis des Hauses kalkulieren, er speichert und öffnet Varianten. Das Programm muss den Plan speichern und öffnen können. Objekte müssen markiert, Eigenschaften müssen geändert werden können. Es muss eine Kalkulation erstellt werden können. Elemente müssen aus Listen ausgewählt werden können.	8	8	
b)	<p>Ausgehend von den konkreten Klassen muss es möglich sein, einen allgemeinen Fall abzuleiten. Es müssen Klassen vorliegen, die auf gleiche Nachrichten reagieren, deren zugehörige Methoden aber verschieden sind. Zwischen den Klassen muss ein Merkmal angegeben werden können, dass die Klassen unterscheidbar macht. Da jetzt gleiche Methoden in die Super-Klasse ausgelagert werden können, wird Redundanz vermieden.</p> <p>Neben der Vererbung muss Delegation als Alternative in Betracht gezogen werden. Bei diesem Konstrukt benutzt eine Klasse eine andere Klasse. Delegation führt in der Client-Klasse zu sehr kurzen Methoden, da die Arbeit an ein anderes Objekt i.d.R. einer anderen Klasse übergeben wird.</p> <p>Vererbung darf nur eingesetzt werden, wenn die Sub-Klassen Spezialfälle der Super-Klasse sind. Vererbung setzt i. A. Abstraktion voraus. Damit wird es möglich, nicht auf den konkreten Details zu programmieren, um von den Änderungen an kapselbaren Entwurfsentscheidungen wenig betroffen zu sein. – Lose Kopplung)</p>	4	8	6
c)	<p>Das Haus hat zwei Stockwerke, in jedem Stockwerk gibt es mehrere Wände, eine Wand kann eine Außenwand und Innenwand sein. Eine Innenwand kann nur Türen enthalten, Außenwände enthalten Fenster und Türen.</p> <p>Innen- bzw. Außenwand sind ein Spezialfall einer allgemeinen (abstrakten) Wand.</p> <p>Fenster und Türen sind in Wänden enthalten. Wände, Fenster und Türen sind Bauelemente.</p> <p>Hier ist ein übergeordnetes abstraktes Element gegeben. So können später beispielsweise leichter andere Elemente wie eine Durchreiche ergänzt werden.</p> <p>Es sind Alternativen denkbar. So könnte beispielsweise die Factory-Methode verwendet werden.</p>			
			10	10

	Lösungsskizze	I	II	III
d)	<p>Es könnte das Array oder eine Collection: LinkedList, ArrayList, Vector verwendet werden.</p> <p>Ein Array vom Typ BauElement ist die einfachste Form, aber statisch. Das Programm legt die maximale Anzahl von Elementen fest. Der Zugriff auf die einzelnen Elemente des Feldes erfolgt über einen Index. In Arrays können auch einfache Datentypen gespeichert werden. – Hier haben sie auch ihre Berechtigung. Für Klassen sollte eine Collection eingesetzt werden.</p> <p>Collections sind flexibler und können nur Objekte enthalten. Einfache Datentypen müssten mit einem »Wrapper« eingepackt werden. Die Anzahl der enthaltenen Elemente passt sich während der Programmlaufzeit an die Erfordernisse an. Der Zugriff auf die Elemente der Collection erfolgt über einen Iterator. Beide Elemente sind in der Java-Bibliothek enthalten.</p> <p>Eine LinkedList könnte die Reihenfolge an einander gefügten Elemente wiedergeben.</p>	20		
e)	<p>Der GesamtPreis kann wie folgt kalkuliert werden,</p> <pre>import java.util.*; public class Geschoss{ private ArrayList bauElemente; public Preis gibGesamtpreis(){ Preis gesamtPreis; BauElement tmpBauElement; ListIterator it = bauElemente.listIterator(bauElemente.size()); while(it.hasPrevious()) { tmpBauElement = (BauElement)it.previous(); myPreis += tmpBauElement.getPreis(); } return gesamtPreis; } }</pre> <p>Die Prüflinge können anstelle des Iterators auch eine for-Schleife nutzen.</p>		10	4
f)	<p>Die Stockwerke müssen eine Referenz auf das über- bzw. unter ihnen liegende Stockwerk besitzen. Nur so können sie sich gegenseitig benachrichtigen. Die Verschiebung der Treppe hat dann in dem einen Stock zur Aufgabe, die Nachricht an das andere Stockwerk zu senden. Damit würde man ein Beobachtermuster nutzen. Es hängt aber davon ab, wie die Koordinatensysteme und die Bezüge im Haus realisiert sind.</p>		6	6
	Insgesamt 100 BWE	32	42	26

Es soll ein einfaches CAD-Programm entwickelt werden, das bei der Planung des Baus von Ein- und Zweifamilienhäusern aus Fertigbauelementen hilft. Dieses Programm soll Käufern zur Verfügung gestellt werden, damit sie sich ihr Traumhaus planen können. Es soll erst einmal nur der Grundriss gezeichnet werden können. Dabei müssen die Bauherren jeweils aus einem bestehenden Katalog vorgegebener Elemente auswählen. In dem Katalog sind die Fertigbauelemente enthalten. Es gibt beispielsweise mehrere Außen- und Innenwände, Türen, Fenster und Treppen. Da die Wandelemente auch transportiert werden müssen, gibt es sie in 5 Längen zwischen 2 m und 4 m. Längere Wände müssen aus Teilwänden zusammengesetzt werden.

Die Skizzen in der Anlage (s. Grundkursaufgabe) zeigen den Grundriss einer zweigeschossigen Wohnung. Es soll mit dieser Skizze nur gezeigt werden, wie beispielsweise Treppen, Türen, Wände und Fenster dargestellt werden.

- Beschreiben Sie typische Interaktionen des Anwenders mit dem Programm. Entwickeln Sie daraus grundlegende Anforderungen an das Programm.
- Geben Sie Kriterien an, wann der Einsatz von Vererbung bei der objektorientierten Modellierung angemessen ist. Beschreiben Sie eine mögliche Alternative zur Vererbung.
- Beschreiben Sie in kurzen Aussagensätzen die wichtigen Teile eines Hauses und geben sie dabei ihre Beziehung an. Entwickeln Sie für dieses System ein Klassendiagramm. Geben Sie wesentliche Attribute und Methoden an. Begründen Sie ausführlich Ihre Entscheidungen bei der Modellierung.
- Alle Elemente eines Geschosses müssen gemeinsam verwaltet werden. In Java gibt es dafür mehrere Möglichkeiten. Beschreiben und vergleichen Sie drei von diesen.
- Gehen Sie jetzt davon aus, dass alle Elemente eines Geschosses mit Hilfe einer ArrayList verwaltet werden und jedes Element über eine Methode gibPreis() verfügt. Implementieren Sie eine Methode gibGesamtpreis und erläutern Sie Ihre Implementation.
- Eine Treppe gehört jeweils zu zwei Geschossen. Nun wird die Treppe im Erdgeschoss verschoben. Dieses muss im Obergeschoss berücksichtigt werden. Entwickeln Sie einen Lösungsansatz für dieses Problem unter Verwendung von Nachrichten zwischen den beteiligten Instanzen.

Kommentar

Es ist sinnvoll, die Bewertung des Anwendungskontextes als weiteren Aufgabenteil zu ergänzen. Dazu müsste ein aktueller „authentischer“ Text vorgegeben werden (z.B. von der Architektenkammer über Selbstplanung von Häusern), zu dem die Prüflinge Stellung nehmen müssten.

Unterrichtliche Voraussetzungen

Die Aufgabe setzt voraus, dass sich die Prüflinge die im Rahmenplan genannten Konzepte der objektorientierten Modellierung und Programmierung erarbeitet haben.

Dazu gehört u. a. die Erstellung von Use-Case- und Klassendiagrammen (UML), die Eigenschaften verschiedenartiger Beziehungen zwischen Klassen sowie die Konzepte der Kapselung, der Vererbung und der Polymorphie. Ein Schwerpunkt stellte die Beurteilung wichtiger Modellierungsalternativen dar, dazu gehört beispielsweise die Alternative Vererbung / Delegation. Modelle sind mit Hilfe Java implementiert und Klassenbibliotheken genutzt worden. In Java wurden die einfachen Datentypen, der strukturierte Datentyp Array und einfache Konzepte der Collections behandelt. Darüber hinaus wurden einige Entwurfsmuster, u.a. das Kompositum erarbeitet.

Im Zusammenhang mit der Implementation von Methoden für Grafikobjekte wurde das Drehen behandelt. Bezüglich der Aufgabenstellung d) wurden mindestens drei Varianten im Unterricht diskutiert.

Hilfsmittel: Dokumentationen von Klassenbibliotheken

Erwartungshorizont

	Lösungsskizze	I	II	III
a)	<p>Der Anwender erstellt Geschosse, wählt, positioniert, verschiebt, dreht, löscht Bauelemente, er lässt den Plan zeichnen, er lässt den Preis des Hauses kalkulieren, er speichert und öffnet Varianten. Das Programm muss den Plan speichern und öffnen können. Objekte müssen markiert, Eigenschaften müssen geändert werden können. Es muss eine Kalkulation erstellt werden können. Elemente müssen aus Listen ausgewählt werden können.</p>	8	10	
b)	<p>Ausgehend von den konkreten Klassen muss es möglich sein, einen allgemeinen Fall abzuleiten. Es müssen Klassen vorliegen, die auf gleiche Nachrichten reagieren, deren zugehörige Methoden aber verschieden sind. Zwischen den Klassen muss ein Merkmal angegeben werden können, das die Klassen unterscheidbar macht. Da jetzt gleiche Methoden in die Super-Klasse ausgelagert werden können, wird Redundanz vermieden.</p> <p>Neben der Vererbung muss Delegation als Alternative in Betracht gezogen werden. Bei diesem Konstrukt benutzt eine Klasse eine andere Klasse. Delegation führt in der Client-Klasse zu sehr kurzen Methoden, da die Arbeit an ein anderes Objekt i.d.R. einer anderen Klasse übergeben wird.</p> <p>Vererbung darf nur eingesetzt werden, wenn die Sub-Klassen Spezialfälle der Super-Klasse sind. Vererbung setzt i. A. Abstraktion voraus. Damit wird es möglich, nicht auf den konkreten Details zu programmieren, um von den Änderungen an kapselbaren Entwurfsentscheidungen wenig betroffen zu sein. – Lose Kopplung)</p>	4	10	6
c)	<p>Das Haus hat zwei Stockwerke, in jedem Stockwerk gibt es mehrere Wände, eine Wand kann eine Außenwand und Innenwand sein. Eine Innenwand kann nur Türen enthalten, Außenwände enthalten Fenster und Türen.</p> <p>Innen- bzw. Außenwand sind ein Spezialfall einer allgemeinen (abstrakten) Wand.</p> <p>Fenster und Türen sind in Wänden enthalten. Wände, Fenster und Türen sind Bauelemente.</p> <p>Hier ist ein übergeordnetes abstraktes Element gegeben. So können später beispielsweise leichter andere Elemente wie eine Durchreiche ergänzt werden.</p> <p>Es sind Alternativen denkbar. So könnte beispielsweise die Factory-Methode verwendet werden.</p>		12	10

	Lösungsskizze	I	II	III
d)	Die ArrayList gehört zu den Collections, die die flexible Handhabung einer großen Anzahl von Objekten erlaubt. Die Anzahl der enthaltenen Elemente passt sich während der Programmlaufzeit an die Erfordernisse an. Zu einer Collection kann jeder Zeit ein weiteres Element hinzugefügt werden. Der Zugriff auf die Elemente der Collection erfolgt über einen Iterator. Beide Elemente sind in der Java-Bibliothek enthalten.	20		
e)	<p>Der GesamtPreis kann wie folgt kalkuliert werden,</p> <pre>import java.util.*; public class Geschoss{ private ArrayList bauElemente; public Preis gibGesamtpreis(){ Preis gesamtPreis; BauElement tmpBauElement; ListIterator it = bauElemente.listIterator(bauElemente.size()); while(it.hasPrevious()) { tmpBauElement = (BauElement)it.previous(); myPreis += tmpBauElement.getPreis(); } return gesamtPreis; } }</pre> <p>Die Prüflinge können anstelle des Iterators auch eine for-Schleife nutzen.</p>			
	Insgesamt 100 BWE	32	42	26

3.3 Kryptographie

GK

Lesen Sie den Auszug aus der Pressemitteilung des Landesbeauftragten für den Datenschutz Schleswig-Holstein (siehe Anlage).

- a) Beschreiben Sie die Funktionsweise eines Kryptosystems mit öffentlichen und privaten Schlüsseln am Beispiel einer E-Mail eines Bürgers an den Datenschutzbeauftragten und der Antwort des Datenschutzbeauftragten.
- b) In der Pressemitteilung steht, dass die Verschlüsselung zu übermittelnder Daten in Deutschland gesetzlich nicht beschränkt ist. Stellen Sie Argumente für und gegen eine gesetzliche Beschränkung von Verschlüsselung gegenüber. Welche Position vertreten Sie selbst? Begründen Sie.
- c) Erläutern Sie die Begriffe Vertraulichkeit und Authentizität im Rahmen einer sicheren Kommunikation. Was müssen der Bürger und der Datenschutzbeauftragte tun, um die Authentizität sicherzustellen?
- d) Wie schätzen Sie die Verbreitung von Verschlüsselungsverfahren, die er mit dieser Mitteilung 1997 fördern wollte, heute im privaten, geschäftlichen und öffentlichen Bereich ein? Geben Sie Gründe für die Verwendung bzw. Nichtverwendung an.
- e) Beschreiben Sie je ein historisches Substitutions- und Transpositionsverfahren. Warum stellen historische Verfahren keine Grundlage für heutige Verschlüsselungsprogramme dar?
- f) Das Verschlüsselungsprogramm PGP verwendet das RSA-Verfahren. Worauf beruht die Sicherheit des RSA-Verfahrens?
- g) Zeigen sie beispielhaft anhand der Zahlen $p = 13$ und $l = 17$ die Schlüsselerzeugung von n , e und d , wobei $e = 7$ sein soll. Verschlüsseln Sie die Nachricht $m = 5$.

Anlage

Der Landesbeauftragte für den Datenschutz Schleswig-Holstein

28. Juli 1997

PRESSEMITTEILUNG

Schleswig-Holsteinischer Datenschutzbeauftragter nimmt ab sofort E-Mails auch verschlüsselt entgegen

Wer in offenen Netzen wie z. B. dem Internet Informationen austauschen will, kann sich gegen Mithören und Mitlesen durch unbefugte Dritte wirksam nur durch die Verschlüsselung der zu übermittelnden Daten schützen. Dies ist in Deutschland gesetzlich in keiner Weise beschränkt. Deshalb empfiehlt der Schleswig-Holsteinische Datenschutzbeauftragte, hiervon Gebrauch zu machen, wann immer es möglich ist.

Da sich Bürger immer häufiger auch per E-Mail an ihn wenden, hat er ab sofort die Voraussetzungen dafür geschaffen, dass dies in verschlüsselter Form möglich ist. Dabei wird das Verschlüsselungsprogramm „Pretty Good Privacy“ (PGP) angewandt, das nach heutigem Kenntnisstand eine gute Sicherheit bietet und kostenlos aus dem Internet entnommen werden kann.

Das Verfahren funktioniert so: Der Datenschutzbeauftragte gibt jedem, der Interesse daran hat, seinen „öffentlichen Schlüssel“ bekannt. Wer mit ihm vertraulich kommunizieren will, kann seinen Text vor der Absendung mit diesem „public key“ verschlüsseln. Die Entschlüsselung ist nur dem Datenschutzbeauftragten mit seinem nur ihm bekannten „private key“ möglich.

Der Datenschutzbeauftragte will auf diesem Wege die Verbreitung von Verschlüsselungsverfahren fördern und selbst Erfahrungen beim Umgang mit der Verschlüsselungstechnik sammeln.

Unterrichtliche Voraussetzungen

Im Unterricht wurden historische Substitutions- und Transpositionsverfahren, symmetrische und asymmetrische Verschlüsselungsverfahren behandelt. Bezüglich des RSA-Verfahrens sind die Algorithmen zur Schlüsselerzeugung, zum Ver- und Entschlüsseln von Nachrichten und zur Authentifizierung bekannt und geübt. Die Prüflinge kennen Verfahren zur Kryptoanalyse der historischen Verfahren und die Komplexität des Faktorisierungsproblems. Aktuelle Einsatzbereiche von Kryptographie sind bekannt und diskutiert worden.

Hilfsmittel: Taschenrechner

Erwartungshorizont

	Lösungsskizze	I	II	III
a)	Ein Bürger verschlüsselt seine Nachricht an den Datenschutzbeauftragten mit dessen öffentlichen Schlüssel. Die verschlüsselte Nachricht kann nur mit dem zugehörigen privaten Schlüssel, den der rechtmäßige Empfänger besitzt, entschlüsselt werden. Diese Nachricht sollte auch den öffentlichen Schlüssel des Absenders enthalten, damit der Datenschutzbeauftragte ebenfalls seine Antwort an den Bürger verschlüsseln kann und nur der gewünschte Empfänger sie entschlüsseln kann.	8	8	
b)	Als Argument für eine gesetzliche Beschränkung wird oft die dem Staat prinzipiell zu ermöglichende Entschlüsselung im Rahmen der Kriminalitätsbekämpfung genannt. Gegen diese Beschränkung spricht das Grundrecht des Brief-, Post- und Fernmeldegeheimnisses. Formulierung und Begründung einer eigenen Position		6	8
c)	„Vertraulichkeit“ bedeutet, dass nur die Personen, an die eine Nachricht gerichtet ist, diese Nachricht lesen können. „Authentizität“ bedeutet die Echtheit, Glaubwürdigkeit einer Nachricht. Die Authentizität einer Nachricht wird dadurch sichergestellt, dass der Absender seine Nachricht zusätzlich mit seinem privaten Schlüssel signiert. Der Empfänger kann dann durch Anwendung des öffentlichen Schlüssels des Absenders verifizieren, ob die Nachricht wirklich von ihm stammt.	8	8	
d)	Erwartet werden Aussagen zu Schutzbedürftigkeit, Verfügbarkeit und Aufwand. Im privaten Bereich werden E-Mails in der Regel aus Bequemlichkeitsgründen und geringerer Schutzbedürftigkeit nicht verschlüsselt. Im geschäftlichen Bereich werden beispielsweise Banken im Rahmen des Online-Bankings mit sicheren Verschlüsselungen. Im öffentlichen Bereich ist die Authentizität besonders wichtig.			12
e)	Die Caesar-Verschlüsselung ist ein Substitutionsverfahren: jeder Buchstabe des Klartextes wird durch einen im Alphabet im festen Abstand folgenden Buchstaben ersetzt. Schon bei relativ kurzen Texten kann mit Hilfe der Buchstabenhäufigkeit zunächst der Buchstabe „E“ und damit auch alle anderen entschlüsselt werden. Die Skytale von Sparta ist ein Transpositionsverfahren. Die Buchstaben des Klartextes bleiben erhalten und werden nur in einer anderen Reihenfolge angeordnet. Durch Aufspüren der häufigen Zweierkombinationen von Buchstaben kann die Umordnung der Buchstaben ermittelt werden.	12	12	
f)	Die Schlüssel werden als Produkt von sehr großen Primzahlen gebildet. Die Umkehrung, nämlich die Faktorisierung des Produktes, ist praktisch unmöglich.	6		

	Lösungsskizze	I	II	III
g)	<p>Es ist $n = p \cdot q = 13 \cdot 17 = 221$ und $(p - 1)(q - 1) = 12 \cdot 16 = 192$.</p> <p>Wegen $1 = 55 \cdot 7 - 2 \cdot 192$ ist dann $d = 55$.</p> <p>Verschlüsselung: Wegen $57 = 78125 = 353 \cdot 221 + 112$ wird 5 zu 112 verschlüsselt.</p>		6	6
	Insgesamt 100 BWE	34	40	26

3.4 Kryptographie

LK

Lesen Sie den Auszug aus der Pressemitteilung des Landesbeauftragten für den Datenschutz Schleswig-Holstein (**siehe Anlage zur Grundkursaufgabe**).

- a) Beschreiben Sie die Anwendung der asymmetrischen Verschlüsselung am Beispiel der E-Mail-Kommunikation mit dem Datenschutzbeauftragten.
- b) Beschreiben Sie das RSA-Verfahren und erläutern Sie an Hand eines Beispiels die Schlüsselerzeugung sowie die Ver- und Entschlüsselung.
- c) Zur Gewinnung sehr großer Zahlen, die mit hoher Wahrscheinlichkeit Primzahlen sind, erzeugt man zunächst Zufallszahlen und testet diese auf Primzahleigenschaft. Für den Test kann der kleine Satz von Fermat genutzt werden:

Ist p eine Primzahl, so gilt für jede natürliche Zahl $b < p$ $b^{p-1} \bmod p = 1 \bmod p$.

p ist somit sicher zerlegbar, wenn eine Zahl $b < p$ gibt, so dass $b^{p-1} \bmod p \neq 1 \bmod p$. Je mehr Zahlen $b < p$ gefunden werden können, welche die obige Gleichung erfüllen, umso größer ist die Wahrscheinlichkeit, dass p prim ist.

Entwickeln Sie einen Primzahltest auf der Basis des kleinen Satzes von Fermat.

- c1) Geben Sie zunächst einen Algorithmus an, der eine große Zufallszahl mit Hilfe Ihres Testes auf Primzahleigenschaft testet.
- c2) Implementieren Sie diesen Algorithmus.
- d) Vergleichen Sie symmetrische und asymmetrische Verschlüsselungsverfahren und deren Einsatzbereiche. Bewerten Sie beide hinsichtlich Anwendbarkeit und Sicherheit.
- e) Diskutieren Sie eine gesetzliche Beschränkung von Kryptographie und beziehen Sie begründet Stellung.

Unterrichtliche Voraussetzungen

Im Unterricht wurden symmetrische und asymmetrische Verschlüsselungsverfahren behandelt. Bezüglich des RSA-Verfahrens sind die Algorithmen zur Schlüsselerzeugung, zum Ver- und Entschlüsseln von Nachrichten und zur Authentifizierung bekannt, geübt und implementiert worden. Die Prüflinge sind mit der Langzahlarithmetik und dem Rechnen modulo n vertraut, hingegen ist der Fermat-Test nicht bekannt. Sie kennen Verfahren zur Kryptoanalyse und die Komplexität des Faktorisierungsproblems. Aktuelle Einsatzbereiche von Kryptographie sind im Unterricht behandelt und diskutiert worden.

Hilfsmittel: Taschenrechner

3.5 Scotland Yard

GK

Bei dem Spiel Scotland Yard jagen mehrere Detektive in London einen Mister X. Einer der Spieler ist Mister X und versteckt sich auf seiner Flucht kreuz und quer durch London. Er zieht unsichtbar und zeigt sich nur in bestimmten Abständen. Alle anderen Spieler sind die Detektive von Scotland Yard und sind hinter ihm her, um ihn zu finden. Gelingt es einem Detektiv mit dem unsichtbaren Mister X auf einem Punkt zusammenzutreffen, muss sich Mister X zeigen und die Detektive haben gewonnen.

Die Detektive haben 22 Runden Zeit, um Mister X zu fangen. Während dieser 22 Runden muss der Detektiv sich nur vier Mal zeigen. Die Detektive haben 10 Taxi-, 8 Bus- und 4-U-Bahn-Tickets, die sie geschickt einsetzen müssen. In einer Runde kann immer nur ein Ticket eingesetzt werden. Mister X verfügt über beliebig viele Tickets und hat auch ein paar Sondertickets.

Der Spielplan zeigt die zulässigen Taxi-, Bus- und U-Bahn-Verbindungen. Die Abbildung 1 zeigt einen kleinen Ausschnitt vom *rechten* Rand des Spielplans.

Die Detektive planen ihr gemeinsames Vorgehen. Sie vermuten Mister X in der Nähe von Haltestelle 55. Ein Teil der Absprache ist, dass Detektiv Blau von der Haltestelle 160 (Start) zur Haltestelle 71 (Ziel) auf der anderen Seite der Themse fahren soll. Alle Detektive machen sich Gedanken über die günstigste Verbindung.

- *Rot* möchte, dass Blau in jeder Runde *möglichst dicht an das Ziel* herankommt.
 - *Grün* schlägt vor, dass Blau die Verbindung wählt, bei der man mit der *geringsten Anzahl von Runden* zum Ziel gelangt.
 - *Blau* möchte die unterschiedliche Anzahl der Tickets berücksichtigen. Dazu ordnet er ihnen Kosten zu (Taxi: 3 BE; Bus: 4 BE; U-Bahn: 9 BE – BE: Beförderungseinheiten). Er möchte die Verbindung nutzen, bei der die insgesamt *notwendigen Beförderungseinheiten am kleinsten* sind.
 - *Gelb* hält sich aus der Diskussion erst einmal heraus.
- a) Zeichnen Sie den Graphen auf Grundlage der *Abbildung 1* und der *Tabelle 1* entsprechend den Vorstellungen von *Blau*.
- b) Wenden Sie den Dijkstra-Algorithmus auf den Graphen von Aufgabenteil a) an. Stellen Sie die einzelnen Schritte tabellarisch dar.
- c) Bestimmen Sie die optimale(n) Verbindung(en) entsprechend den Vorstellungen von *Rot* und *Grün*.
- d) *Rot* und *Grün* wenden allgemeine Suchverfahren an. Benennen Sie die verwendeten Verfahren und beschreiben Sie das von *Rot* benutzte Verfahren.
- e) Erstellen Sie eine Tabelle in der unten angegebenen Form und führen Sie die Vor- und Nachteile der einzelnen Vorschläge auf.

Vorschlag	Tickets			Runden	BE
	Taxi	Bus	U-Bahn		
<i>Rot</i>					
<i>Grün</i>					
<i>Blau</i>					

Während der Diskussion über die drei Vorschläge gibt *Gelb* zu Bedenken, dass sich Mister X nach zwei Runden wieder „zeigen“ muss. *Gelb* möchte dies bei der Diskussion der Vorschläge berücksichtigen – Mister X könnte eventuell gar nicht in der Nähe der Haltestelle 55, sondern in einer anderen Ecke von London „auftauchen“.

- f) Erläutern Sie die Befürchtungen von *Gelb* anhand der drei Vorschläge

Materialien

Abbildung 1:
Spielplanausschnitt

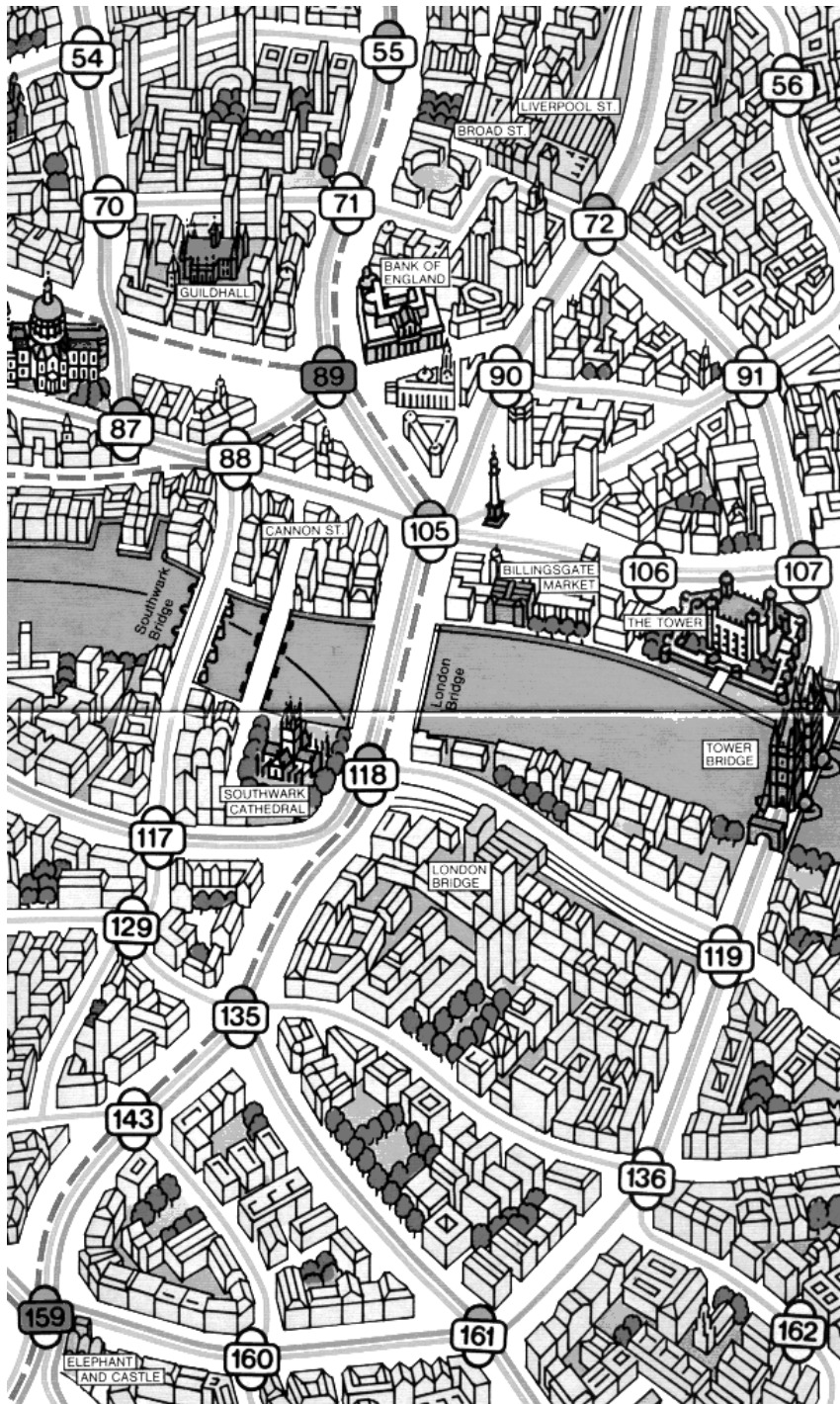


Tabelle 1:

Linien, die berücksichtigt werden sollen

71 mit Taxi nach	72
mit Taxi nach	89
72 mit Taxi nach	91
mit Bus nach	105
mit Bus nach	107
89 mit Taxi nach	105
mit U-Bahn nach	159
91 mit Taxi nach	105
mit Taxi nach	107
105 mit Bus nach	107
mit Taxi nach	118
107 mit Taxi nach	119
mit Bus nach	161
118 mit Taxi nach	119
mit Bus nach	135
135 mit Taxi nach	143
mit Bus nach	59
mit Taxi nach	161
143 mit Taxi nach	159
mit Taxi nach	160
159 mit Taxi nach	160
mit Bus nach	161
160 mit Taxi nach	161

Tabelle 2:

Die **Luftlinienentfernungen** in Pixeln
(Startknoten Zielknoten Entfernung):

(71 71 000)	(72 71 183)	(89 71 126)	(91 71 320)
(105 71 150)	(107 71 426)	(118 71 414)	(119 71 611)
(135 71 595)	(143 71 681)	(159 71 835)	(160 71 840)
(161 71 826)			

Angaben über die unterrichtlichen Voraussetzungen

Die Schüler müssen unterschiedliche Strategien bei der Suche in Bäumen und (bewerteten) Graphen kennen und sicher anwenden können. Die Fähigkeit, Algorithmen angemessen beschreiben zu können, ist ebenso nötig wie die übersichtliche Darstellung konkreter Suchabläufe. Die Implementation von Suchstrategien mit Hilfe z.B. von Scheme ist nicht zwingend nötig, kann aber das Verständnis für die unterschiedlich zu führenden Listen beispielsweise beim Dijkstra-Algorithmus absichern.

Die Aufgabenstellung geht davon aus, dass im Unterricht Folgendes erarbeitet wurde:
Listen, rekursive Funktionsaufrufe; Suchbaum, Tiefen- und Breitensuche, Backtracking; Graphensuche (best-first-Strategie, Greedy-Algorithmus (gierige Strategie), brute-force-Strategie, Dijkstra-Algorithmus); Vergleich der Suchstrategien.

Hilfsmittel: (wissenschaftlicher) Taschenrechner, Farbkopie des Spielplanausschnitts

Erwartungshorizont

	Lösungsskizze	I	II	III
a)	<p>Dieser Aufgabenteil bildet die Grundlage für Aufgabenteil b), deshalb müssen die Schüler in der Anfertigung derartiger Graphen sicher sein.</p>	8	7	
b)	<p>Die Tabelle (siehe nächste Seite) soll zeigen, dass die Schülerinnen und Schüler den Algorithmus verstanden haben. Das wird deutlich an den Entscheidungen, welche der nachfolgenden Kanten aus der NEXT-Liste in die OPEN-Liste eingefügt werden bzw. dort existierende Kanten ersetzen und welche Kante aus der OPEN-Liste in die CLOSED-Liste verschoben wird.</p> <p>Die Schülerinnen und Schüler können auch eine dreispaltige Darstellung (STEP, CLOSED, OPEN) verwenden, bei der erst einmal alle Nachfolgerkanten in die OPEN-Liste eingefügt und erst dann mit den in der CLOSED- und OPEN-Liste vorhandenen Kanten verglichen werden. Der direkte Rückweg wird in der Regel nicht mit aufgeschrieben.</p>	5	15	5
c)	<p><u>Rot</u>: 160 – 143 – 135 – 118 – 105 – 89 – 71</p> <p><u>Grün</u>: 160 – 159 – 89 – 71</p>	5	10	

	Lösungsskizze				I	II	III
zu b)	STEP	CLOSED	NEXT	OPEN			
	0			(160 160 0) <i>STEP 1</i>			
	1	(160 160 0)	(160 143 3) (160 159 3) (160 161 3)	(160 143 3) <i>STEP 2</i> (160 159 3) <i>STEP 3</i> (160 161 3) <i>STEP 4</i>			
	2	(160 143 3)	(143 135 6) (143 159 6) <i>siehe OPEN</i>	(143 135 6) <i>STEP 5</i>			
	3	(160 159 3)	(159 89 12) (159 135 7) <i>siehe OPEN</i> (159 143 6) <i>siehe CLOSED</i> (159 161 7) <i>siehe OPEN</i>	(159 89 12) <i>STEP 12</i>			
	4	(160 161 3)	(161 107 7) (161 135 6) <i>siehe OPEN</i> (161 159 7) <i>siehe CLOSED</i>	(161 107 7) <i>STEP 6</i>			
	5	(143 135 6)	(135 118 10) (135 159 10) <i>siehe CLOSED</i> (135 161 9) <i>siehe CLOSED</i>	(135 118 10) <i>STEP 7</i>			
	6	(161 107 7)	(107 72 11) (107 91 10) (107 105 11) (107 119 10)	(107 72 11) <i>STEP 10</i> (107 91 10) <i>STEP 8</i> (107 105 11) <i>STEP 11</i> (107 119 10) <i>STEP 9</i>			
	7	(135 118 10)	(118 105 13) <i>siehe OPEN</i> (118 119 13) <i>siehe OPEN</i>				
	8	(107 91 10)	(91 72 13) <i>siehe OPEN</i> (91 105 13) <i>siehe OPEN</i>				
	9	(107 119 10)	(119 118 13) <i>siehe CLOSED</i> (119 136 13) <i>siehe CLOSED</i>				
	10	(107 72 11)	(72 71 14) (72 91 14) <i>siehe CLOSED</i> (72 105 15) <i>siehe OPEN</i>	(72 71 14) <i>STEP 13</i>			
	11	(107 105 11)	(105 72 15) <i>siehe OPEN</i> (105 89 14) <i>siehe OPEN</i> (105 91 14) <i>siehe CLOSED</i> (105 118 14) <i>siehe CLOSED</i>				
	12	(159 89 12)	(89 71 15) <i>siehe OPEN</i> (89 105 15) <i>siehe CLOSED</i>				
	13	(72 71 14)					
160 – 161 – 107 – 72 – 71							

	Lösungsskizze					I	II	III																												
d)	<p><u>Rot</u>: Greedy.</p> <p><u>Grün</u>: Breitensuche (bzw. Dijkstra mit Kantenbewertung 1).</p> <p>Das Greedy-Verfahren verwendet eine Heuristik, die den noch für den Rest der Suche benötigten Aufwand bewertet. Ein sehr anschauliches Beispiel ist die Luftlinienentfernung zum Ziel.</p> <p>Vom derzeitigen Knoten ausgehend werden alle nachfolgenden Knoten ermittelt und derjenige mit den geringsten Kosten ausgewählt – im Beispiel mit der Luftlinienentfernung: dessen Entfernung zum Ziel am kleinsten ist. Von diesem Knoten wird, wenn er nicht der Zielknoten ist, die Suche fortgesetzt.</p> <p><i>Sollte ein Schüler Vor- und Nachteile erwähnen, so sind sie angemessen zu berücksichtigen.</i></p>					6	6	3																												
e)	<table><tr><th rowspan="2">Vorschlag</th><th colspan="3">Tickets</th><th rowspan="2">Runden</th><th rowspan="2">BE</th></tr><tr><th>Taxi</th><th>Bus</th><th>U-Bahn</th></tr><tr><td>Rot</td><td>5</td><td>1</td><td></td><td>6</td><td>19</td></tr><tr><td>Grün</td><td>2</td><td></td><td>1</td><td>3</td><td>15</td></tr><tr><td>Blau</td><td>2</td><td>2</td><td></td><td>4</td><td>14</td></tr></table>	Vorschlag	Tickets			Runden	BE	Taxi	Bus	U-Bahn	Rot	5	1		6	19	Grün	2		1	3	15	Blau	2	2		4	14	<p><i>Die Schüler sollen anhand der Tabelle die Vor- und Nachteile der einzelnen Strategien angeben und dabei insbesondere auf das „schlechte“ Ergebnis beim Greedy-Algorithmus eingehen.</i></p>					6	6	8
Vorschlag	Tickets			Runden	BE																															
	Taxi	Bus	U-Bahn																																	
Rot	5	1		6	19																															
Grün	2		1	3	15																															
Blau	2	2		4	14																															
f)	<p>Der Vorschlag von Blau führt über die Haltestelle 107 (zweite Runde). Von dort aus sind in anderen Teilen Londons liegende Stationen schwieriger zu erreichen. Selbst der in der vergleichenden Bilanz bislang schlechteste Vorschlag von Rot führt über die deutlich besser gelegene Haltestelle 135. Der Vorschlag von Grün führt über die sehr gut gelegene Haltestelle 89 (mit diversen Bus-Verbindungen und U-Bahn-Anschluss).</p>							10																												
	Insgesamt 100 BWE					30	44	26																												

3.6 Scotland Yard

LK

Bei dem Spiel Scotland Yard jagen mehrere Detektive in London einen Mister X. Einer der Spieler ist Mister X und versteckt sich auf seiner Flucht kreuz und quer durch London. Er zieht unsichtbar und zeigt sich nur in bestimmten Abständen. Alle anderen Spieler sind die Detektive von Scotland Yard und sind hinter ihm her, um ihn zu finden. Gelingt es einem Detektiv mit dem unsichtbaren Mister X auf einem Punkt zusammenzutreffen, muss sich Mister X zeigen und die Detektive haben gewonnen.

Die Detektive haben 22 Runden Zeit, um Mister X zu fangen. Während dieser 22 Runden muss der Detektiv sich nur vier Mal zeigen. Die Detektive haben 10 Taxi-, 8 Bus- und 4-U-Bahn-Tickets, die sie geschickt einsetzen müssen. In einer Runde kann immer nur ein Ticket eingesetzt werden. Mister X verfügt über beliebig viele Tickets und hat auch ein paar Sondertickets.

Der Spielplan zeigt die zulässigen Taxi-, Bus- und U-Bahn-Verbindungen. Die Abbildung 1 zeigt einen kleinen Ausschnitt vom *rechten* Rand des Spielplans.

Die Detektive planen ihr gemeinsames Vorgehen. Sie vermuten Mister X in der Nähe von Haltestelle 55. Ein Teil der Absprache ist, dass Detektiv Blau von der Haltestelle 160 (Start) zur Haltestelle 71 (Ziel) auf der anderen Seite der Themse fahren soll. Alle Detektive machen sich Gedanken über die günstigste Verbindung.

- *Rot* möchte, dass Blau in jeder Runde *möglichst dicht an das Ziel* herankommt.
 - *Grün* schlägt vor, dass Blau die Verbindung wählt, bei der man mit der *geringsten Anzahl von Runden* zum Ziel gelangt.
 - *Blau* möchte die unterschiedliche Anzahl der Tickets berücksichtigen. Dazu ordnet er ihnen Kosten zu (Taxi: 3 BE; Bus: 4 BE; U-Bahn: 9 BE – BE: Beförderungseinheiten). Er möchte die Verbindung nutzen, bei der die insgesamt *notwendigen Beförderungseinheiten am kleinsten* sind.
 - *Gelb* hält sich aus der Diskussion erst einmal heraus.
- a) Zeichnen Sie den Graphen auf Grundlage der *Abbildung 4.1* und der *Tabelle 4.1* entsprechend den Vorstellungen von *Blau*.
- b) Beschreiben Sie den Dijkstra-Algorithmus in Worten.
- c) Wenden Sie den Dijkstra-Algorithmus auf den Graphen von Aufgabenteil a) an. Stellen Sie die einzelnen Schritte tabellarisch dar.
- d) Bestimmen Sie die optimale(n) Verbindung(en) entsprechend den Vorstellungen von *Rot* und *Grün*.
- e) *Rot* und *Grün* wenden allgemeine Suchverfahren an. Benenne Sie die verwendeten Verfahren und beschreiben Sie das von *Rot* benutzte Verfahren.
- f) Erstellen Sie eine Tabelle in der unten angegebenen Form und diskutieren Sie anschließend die drei Vorschläge.

<i>Vorschlag</i>	<i>Tickets</i>			<i>Runden</i>	<i>BE</i>
	<i>Taxi</i>	<i>Bus</i>	<i>U-Bahn</i>		
Rot					
Grün					
Blau					

Während der Diskussion über die drei Vorschläge gibt *Gelb* zu Bedenken, dass sich Mister X nach zwei Runden wieder „zeigen“ muss. *Gelb* möchte dies bei der Diskussion der Vorschläge berücksichtigen – Mister X könnte eventuell gar nicht in der Nähe der Haltestelle 55, sondern in einer anderen Ecke von London „auftauchen“.

- g) Erläutern Sie die Befürchtungen von *Gelb* anhand der drei Vorschläge.
- h) Entwickeln Sie eine Suchstrategie, die die Befürchtungen von *Gelb* berücksichtigt.

Materialien

Abbildung 1:
Spielplanausschnitt

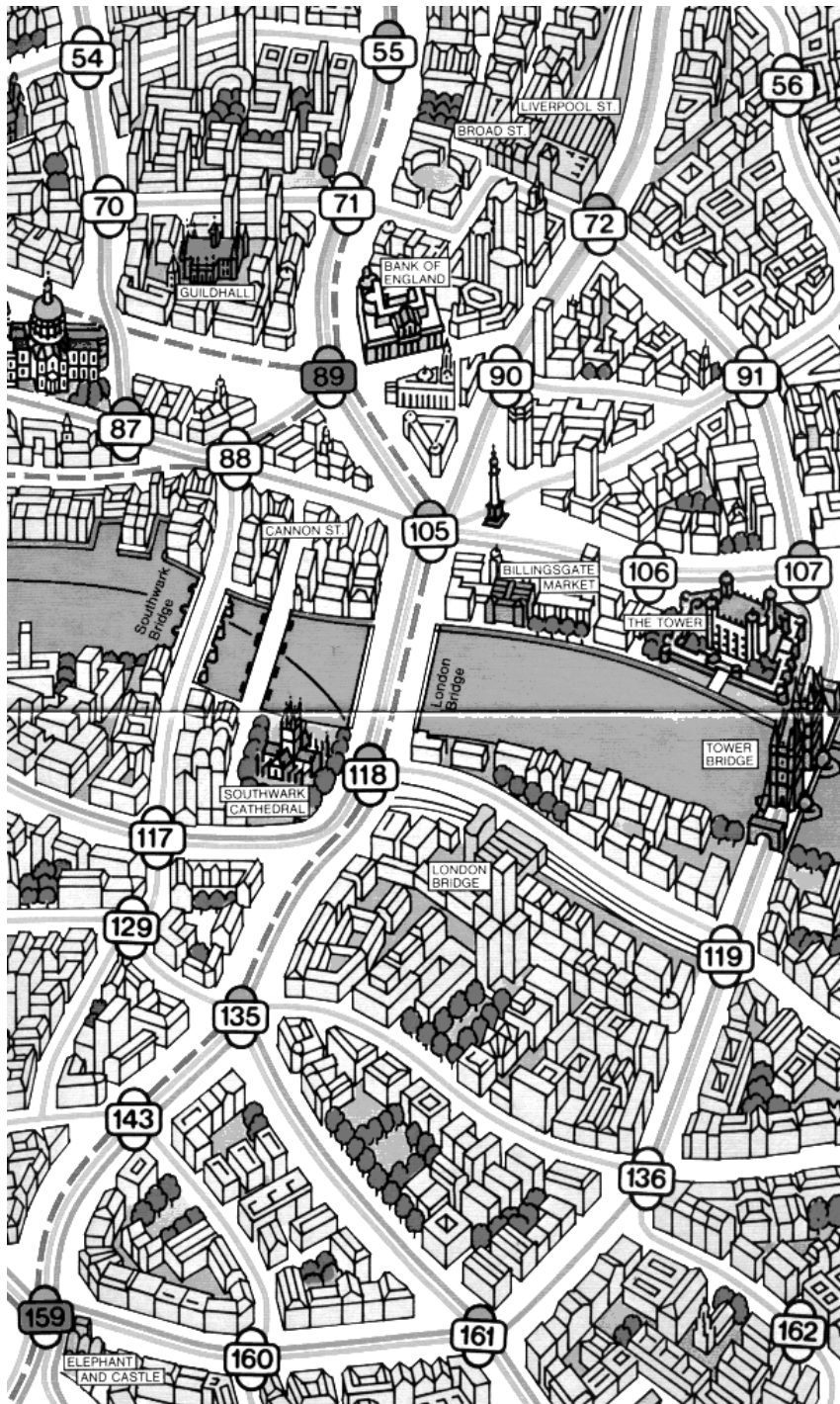


Tabelle 1:

Linien, die berücksichtigt werden sollen

71 mit Taxi nach	72
mit Taxi nach	89
72 mit Taxi nach	91
mit Bus nach	105
mit Bus nach	107
89 mit Taxi nach	105
mit U-Bahn nach	159
91 mit Taxi nach	105
mit Taxi nach	107
105 mit Bus nach	107
mit Taxi nach	118
107 mit Taxi nach	119
mit Bus nach	161
118 mit Taxi nach	119
mit Bus nach	135
119 mit Taxi nach	136
135 mit Taxi nach	136
mit Taxi nach	143
mit Bus nach	59
mit Taxi nach	161
143 mit Taxi nach	159
mit Taxi nach	160
159 mit Taxi nach	160
mit Bus nach	161
160 mit Taxi nach	161

Tabelle 2:

Die **Luftlinienentfernungen** in Pixeln
(Startknoten Zielknoten Entfernung):

(71 71 000)	(72 71 183)	(89 71 126)	(91 71 320)
(105 71 150)	(107 71 426)	(118 71 414)	(119 71 611)
(135 71 595)	(136 71 738)	(143 71 681)	(159 71 835)
(160 71 840)	(161 71 826)		

Angaben über die unterrichtlichen Voraussetzungen

Die Schüler müssen unterschiedliche Strategien bei der Suche in Bäumen und (bewerteten) Graphen kennen und sicher anwenden können. Die Fähigkeit, Algorithmen angemessen beschreiben zu können, ist ebenso nötig wie die übersichtliche Darstellung konkreter Suchabläufe. Die Implementation von Suchstrategien mit Hilfe z.B. von Scheme ist nicht zwingend nötig, kann aber das Verständnis für die unterschiedlich zu führenden Listen beispielsweise beim Dijkstra-Algorithmus absichern.

Die Aufgabenstellung geht davon aus, dass im Unterricht Folgendes erarbeitet wurde:
Listen, rekursive Funktionsaufrufe; Suchbaum, Tiefen- und Breitensuche, Backtracking; Graphensuche (best-first-Strategie, Greedy-Algorithmus (gierige Strategie), brute-force-Strategie, Dijkstra-Algorithmus); Vergleich der Suchstrategien.

Hilfsmittel: (wissenschaftlicher) Taschenrechner, Farbkopie des Spielplanausschnitts

Erwartungshorizont

	Lösungsskizze	I	II	III
a)	<p>Dieser Aufgabenteil bildet die Grundlage für Aufgabenteil b), deshalb müssen die Schüler in der Anfertigung derartiger Graphen sicher sein.</p>	8	6	
b)	<p>Am Anfang enthält die offene Liste (OPEN) die Startkante – hier in der Form (160 160 0). Die geschlossene Liste (CLOSED) ist am Anfang leer.</p> <p>Falls OPEN leer ist, wird die Suche abgebrochen. Sonst wird in OPEN die Kante K mit der kleinsten Kantenbewertung gesucht. Die Kante K wird aus OPEN gestrichen und in CLOSED eingefügt. Wenn der Zielknoten der Kante K gleich dem Ziel ist, kann die Suche mit der Ausgabe des optimalen Weges (ermittelt aus CLOSED) abgeschlossen werden. Wenn das Ziel noch nicht erreicht wurde, werden die Nachfolger (mit aufsummierten Kantenbewertungen) der Kante K bestimmt. Ein Nachfolger wird nicht in OPEN übernommen, wenn</p> <ul style="list-style-type: none"> – in CLOSED eine Kante mit dem gleichen Zielknoten vorkommt oder – in OPEN eine Kante mit dem gleichen Zielknoten vorkommt und deren aufsummierte Kantenbewertung nicht größer ist als bei dem Nachfolger. <p>Danach überprüft die Suchroutine wieder, ob OPEN leer ist. Usw.</p> <p><i>Dieses Verfahren könnte auch im Pseudo-Code dargestellt werden:</i></p> <p>G sei ein ungerichteter Graph mit einer Kostenfunktion $g(U,V)$. Der Startknoten sei A, Zielknoten Z.</p> <ol style="list-style-type: none"> 1. Erzeuge eine leere Liste OPEN (untersuchte Knoten) und eine leere Liste CLOSED (besuchte Knoten). 2. Trage den Startknoten als (A,A,0) in OPEN ein. 3. Wähle aus OPEN eine Kante (U,V,b) mit minimalem b. Ist OPEN leer, existiert kein kürzester Weg; sonst füge (U,V,b) zu CLOSED hinzu und entferne (U,V,b) aus OPEN. Wenn V der Zielknoten Z ist, dann war die Suche erfolgreich. 			

	Lösungsskizze				I	II	III
	<p>4. Bilde die Menge aller Nachfolgekanten $(V, W, b+g(V, W))$, die von V ausgehen. Ist das Ziel $(*, W, *)$ dieser Kante weder in CLOSED noch OPEN, dann füge $(V, W, b+g(V, W))$ zu OPEN hinzu; ist das Ziel $(*, W, b')$ der Kante bereits in OPEN und ist $b+g(U, V) < b'$, dann ersetze $(*, W, b')$ in OPEN durch $(V, W, b+g(V, W))$.</p> <p>5. Gehe zu 3.</p>				6	6	2
c)	<p>Die Tabelle (siehe nächste Seite) soll zeigen, dass die SchülerInnen den Algorithmus verstanden haben. Das wird deutlich an den Entscheidungen, welche der nachfolgenden Kanten aus der NEXT-Liste in die OPEN-Liste eingefügt werden bzw. dort existierende Kanten ersetzen und welche Kante aus der OPEN-Liste in die CLOSED-Liste verschoben wird.</p> <p>Die Schüler können auch eine dreispaltige Darstellung (STEP, CLOSED, OPEN) verwenden, bei der erst einmal alle Nachfolgekanten in die OPEN-Liste eingefügt und erst dann mit den in der CLOSED- und OPEN-Liste vorhandenen Kanten verglichen werden. Der direkte Rückweg wird in der Regel nicht mit aufgeschrieben.</p>				4	15	
d)	<p><u>Rot</u>: 160 – 143 – 135 – 118 – 105 – 89 – 71</p> <p><u>Grün</u>: 160 – 159 – 89 – 71</p>				4	6	
e)	<p><u>Rot</u>: Greedy.</p> <p><u>Grün</u>: Breitensuche (bzw. Dijkstra mit Kantenbewertung 1).</p> <p>Das Greedy-Verfahren verwendet eine Heuristik, die den noch für den Rest der Suche benötigten Aufwand bewertet. Ein sehr anschauliches Beispiel ist die Luftlinienentfernung zum Ziel.</p> <p>Vom derzeitigen Knoten ausgehend werden alle nachfolgenden Knoten ermittelt und derjenige mit den geringsten Kosten ausgewählt – im Beispiel mit der Luftlinienentfernung: dessen Entfernung zum Ziel am kleinsten ist. Von diesem Knoten wird, wenn er nicht der Zielknoten ist, die Suche fortgesetzt.</p> <p><i>Sollte ein Schüler Vor- und Nachteile erwähnen, so sind sie angemessen zu berücksichtigen.</i></p>				4	4	2
zu c)	STEP	CLOSED	NEXT	OPEN			
	0			(160 160 0) <i>STEP 1</i>			
	1	(160 160 0)	(160 143 3) (160 159 3) (160 161 3)	(160 143 3) <i>STEP 2</i> (160 159 3) <i>STEP 3</i> (160 161 3) <i>STEP 4</i>			
	2	(160 143 3)	(143 135 6) (143 159 6) <i>siehe OPEN</i>	(143 135 6) <i>STEP 5</i>			
	3	(160 159 3)	(159 89 12) (159 135 7) <i>siehe OPEN</i> (159 143 6) <i>siehe CLOSED</i> (159 161 7) <i>siehe OPEN</i>	(159 89 12) <i>STEP 13</i>			
	4	(160 161 3)	(161 107 7) (161 135 6) <i>siehe OPEN</i> (161 159 7) <i>siehe CLOSED</i>	(161 107 7) <i>STEP 6</i>			

Lösungsskizze						I	II	III
	5	(143 135 6)	(135 118 10) (135 136 9) (135 159 10) <i>siehe CLOSED</i> (135 161 9) <i>siehe CLOSED</i>	(135 118 10) <i>STEP 8</i> (135 136 9) <i>STEP 7</i>				
	6	(161 107 7)	(107 72 11) (107 91 10) (107 105 11) (107 119 10)	(107 72 11) <i>STEP 11</i> (107 91 10) <i>STEP 9</i> (107 105 11) <i>STEP 12</i> (107 119 10) <i>STEP 10</i>				
	7	(135 136 9)	(136 119 12) <i>siehe OPEN</i>					
	8	(135 118 10)	(118 105 13) <i>siehe OPEN</i> (118 119 13) <i>siehe OPEN</i>					
	9	(107 91 10)	(91 72 13) <i>siehe OPEN</i> (91 105 13) <i>siehe OPEN</i>					
	10	(107 119 10)	(119 118 13) <i>siehe CLOSED</i> (119 136 13) <i>siehe CLOSED</i>					
	11	(107 72 11)	(72 71 14) (72 91 14) <i>siehe CLOSED</i> (72 105 15) <i>siehe OPEN</i>	(72 71 14) <i>STEP 14</i>				
	12	(107 105 11)	(105 72 15) <i>siehe OPEN</i> (105 89 14) <i>siehe OPEN</i> (105 91 14) <i>siehe CLOSED</i> (105 118 14) <i>siehe CLOSED</i>					
	13	(159 89 12)	(89 71 15) <i>siehe OPEN</i> (89 105 15) <i>siehe CLOSED</i>					
	14	(72 71 14)						
160 – 161 – 107 – 72 – 71								
f)	Vorschlag	Tickets		Runden	BE			
		Taxi	Bus	U-Bahn				
	Rot	5	1		6	19		
	Grün	2		1	3	15		
	Blau	2	2		4	14		
Die Schüler sollen anhand der Tabelle die Vor- und Nachteile der einzelnen Strategien vergleichend problematisieren und dabei insbesondere auf das „schlechte“ Ergebnis beim Greedy-Algorithmus eingehen.						5	5	7

	Lösungsskizze	I	II	III
g)	Der Weg von <i>Blau</i> führt über die Haltestelle 107 (zweite Runde). Von dort aus sind in anderen Teilen Londons liegende Stationen schwieriger zu erreichen. Selbst der in der vergleichenden Bilanz bislang schlechteste Weg von <i>Rot</i> führt über die deutlich besser gelegene Haltestelle 135. Der Weg von <i>Grün</i> führt über die sehr gut gelegene Haltestelle 89 (mit diversen Bus-Verbindungen und U-Bahn-Anschluss).			8
h)	Beispielsweise sind folgende Varianten denkbar: <ul style="list-style-type: none"> – Es wird eine Haltestelle (z.B. 89, 118 oder 135) vorgegeben, über die die optimale Verbindung verlaufen muss. Die Suche könnte dadurch in zwei voneinander unabhängige Abschnitte aufgeteilt werden. – Jede Haltestelle erhält je nach Lage und/oder Anschlussreichtum einen sogenannten Ortszuschlag – beispielsweise werden bei Haltestelle 89 null Punkte und bei Haltestelle 136 drei Punkte zu den Kosten hinzugezählt. 			8
	Insgesamt 100 BWE	31	42	27

Vorsatzblatt (Vorderseite) :

Schulstempel

Nummer des Umschlags: _____

Vorsatzblatt für jede Aufgabe

1. Angaben durch die Schulleitung

Nummer der Aufgabe: _____

LK/GK _____

Fach: _____

(nur für Bildende Kunst)

Arbeitszeit: _____

(nur bei Fremdsprachen)

Unterricht ab Klasse: _____

2. Angaben des Fachlehrers / der Fachlehrerin im 3. Halbjahr (Referent / in)

(ggf. in Absprache mit dem Fachlehrer / der Fachlehrerin im 1. Halbjahr und / oder 2. Halbjahr)

Referent / in: _____

Aufgabe (Kurzbezeichnung): _____

besondere Hilfsmittel: _____

Halbjahresangabe: _____

Kursthema in dem betr. Halbjahr: _____

ggf. Fachlehrer / in im 1. und / oder 2. Halbjahr _____

3. Angaben über unterrichtliche Voraussetzungen

(auch über Besonderheiten der Kurszusammensetzung o. ä., inhaltliche Schwerpunkte, benutzte Materialien / Bücher)