

Prüfungsvorbereitung

IKA

Programmiertechnik Technik und Theorie

Autoren:

Michael Litera
Jürgen Funke

V 1.0

Inhaltsverzeichnis

1	Grundlagen.....	4
1.1	Fragen	4
1.2	Beispiele	4
2	Ein- und Ausgabe.....	5
2.1	Fragen	5
2.2	Beispiele	5
3	Operatoren.....	6
3.1	Fragen	6
3.2	Beispiele	6
4	Selektion und Iteration	7
4.1	Fragen	7
4.2	Beispiele	7
4.3	Aufgaben	7
5	Funktionen	8
5.1	Fragen	8
5.2	Beispiele	8
5.3	Aufgaben	8
6	Arrays.....	9
6.1	Fragen	9
6.2	Beispiele	9
6.3	Aufgaben	9
7	Zeiger	11
7.1	Fragen	11
7.2	Beispiele	11
7.3	Aufgaben	11
8	Strukturen.....	12
8.1	Fragen	12
8.2	Beispiele	12
8.3	Aufgaben	12
9	Klassen.....	13
9.1	Fragen	13
9.2	Beispiele	13
9.3	Aufgaben	14
10	Dateioperationen	15
10.1	Fragen	15
10.2	Beispiele.....	15

10.3	Aufgaben.....	15
11	Vererbung / Polymorphie	16
11.1	Fragen	16
11.2	Beispiele.....	16
11.3	Aufgaben.....	17
12	PAP.....	18
12.1	Fragen	18
12.2	Beispiele.....	18
12.3	Aufgaben.....	18
13	Struktogramm.....	19
13.1	Fragen	19
13.2	Beispiele.....	19
13.3	Aufgaben.....	19
14	Klassendiagramm	20
14.1	Fragen	20
14.2	Beispiele.....	20
14.3	Aufgaben.....	20

1 Grundlagen

1.1 Fragen

Nr.	Frage	Kapitel
1	Was sind die Bestandteile eines C++ Programmes?	1.2
2	Was ist ein C++ Quell Code und Was bedeutet C++	1.1
3	Was bedeutet Compiler, Linker Bibliotheken	1.3
4	Welche Entwicklungsumgebung kann ich für C++ nutzen?	1.4
5	Konvention für Bezeichner	2.2.2
6	Welches Trennzeichen nutzen wir?	2.2.3
7	Welche Arten von Kommentierung gibt es und wie stehen sie im Zusammenhang mit dem Compiler	2.2.4
8	Welche Arten von Datentypen gibt es und welchen Speicherplatz belegen sie?	2.3
9	Was sind Variablen?	2.3
10	Was bedeutet Deklaration bei Datentypen?	2.3.3

1.2 Beispiele

Nr.	Beispiel	
5	Zahl, _funk, 2*3_dd, 2ma7, break, continue, for1	
7	/* */; //	
8	Int a, double b, float c, char x, bool b,	
8	a = 1i; a = 1.4i; b=2d; c=3f;	

2 Ein- und Ausgabe

2.1 Fragen

Nr.	Frage	Kapitel
1	Welche Bibliothek benötige bei der Ausgabe auf dem Bildschirm oder Eingabe per Tastatur?	3
2	Welche Sonderzeichen habe ich bei der Ausgabe?	
3	Was sind Manipulatoren?	3.1.3
4	Wie gebe ich einen Text aus?	
5	Wie lese ich einen Text ein?	
6	Was erreiche ich mit oct, dec, setprecision (int), setfill (int), setw (int), left right ?	
7	Was ist bei der Eingabe wichtig zu wissen?	
8	Was bezeichnet den Stream Status?	3.2.2

2.2 Beispiele

Nr.	Frage	Kapitel
1	Cin >> a >> b >> c;	
2	\a \b \t \n \" \\ \201	
3	Einfach Beispiel: #####Hallo Hallo#### 1.12 Dezimal: 30 Oktal: 36	
8	Cin.good();	

3 Operatoren

3.1 Fragen

Nr.	Frage	Kapitel
1	Welche elementaren Arten von Operatoren gibt es?	4.1
2	Was sind Arithmetische Operatoren?	
3	Was bedeutet der Modulo Operator?	
4	Was sind Inkrement- und Dekrementoperatoren und was ist hier zu beachten?	
5	Relationale und Logische Operatoren?	
6	Was ist einer Wahrheitstabelle	
7	Bit Operatoren?	
8	Prioritäten von Operatoren?	
9	Was sind gekoppelte Zuweisung?	
10	Was bedeutet der Begriff sizeof?	

3.2 Beispiele

Nr.	Frage	Kapitel
1	Arithmetische, logische, Modulo, Inkrement und Dekrement, Relationale	
2	Welche Ergebnis liefert <code>int a = 1, int b = 1; int c; c = a/b; c = a+b</code>	
3	Wenn ich <code>double c</code> ; Welches Ergebnis dann?	
4	<code>Char a = 'A'; char b = 'B'; char c; c = a+b</code>	
5	<code>Int X=10; ++x; x-- ; x = x+1</code>	
6	Relationale: <code>Int x = 10; (x < 10); X == 10; x != 10;</code>	
7	Logische: <code>int x = 10; (x > 0) && (x < 100), x > 1 x > 2; !x</code>	
8	Bitweise verschieben durch <code>&, , ^, ~</code> z.B. <code>x = x << 2.</code>	
9	<code>Intx; sizeof(x)</code>	
10	<code>X += 10; y /= 5;</code>	
	Prioritäten wichtig entsprechend Rang: z.B. gekoppelte, log, arith, etc	

4 Selektion und Iteration

4.1 Fragen

Nr.	Frage	Kapitel
1	Welche Arten von Selektionen gibt es? Welche Schlüsselworte brauchen Sie dazu	5.1
2	Welche Arten von Iterationen gibt es? Schlüsselworte brauchen Sie dazu	5.1
3	Was bedeuten die Wörter break und continue in einer Iteration und wo kann ich sie nutzen?	

4.2 Beispiele

Nr.	Frage	Kapitel
1	Einseitige (if), zweiseitige (if-else), verschachtelte Selektion (if, if-else), Mehrfachselection mit switch	
2	Kopf (do while), fuss (while), zählergesteuerte (for) Iterationen	
3	Break, continue	

4.3 Aufgaben

- Programmieren Sie ein C++ Programm, das 2 Zahlen vergleicht und überprüft, ob es sich um eine gerade oder ungerade Zahl handelt.
- Programmieren Sie ein C++ Programm, das 2 Zahlen vergleicht und die kleinste Zahl ausgibt.
- Programmieren Sie in C++ eine Mehrfachselektion, die bei den Zahlen 1-5 jeweils die Zahl ausgibt. Sollte keine der Zahlen gewählt werden, geben die Ausgabe „Keine Zahl“ aus dem Bildschirm aus. Sie bleiben nach jeder Eingabe wieder eine neue Eingabe machen. Über die Zahl 0 können sie das Programm beenden.
- Malen Sie ein PAP für jeweils einen Iterationstyp, in dem sie 10 Zahlen ausgeben.

5 Funktionen

5.1 Fragen

Nr.	Frage	Kapitel
1	Was bedeutet der Begriff Funktion und wozu benötigt man diese?	6.
2	Welche Arten von Funktionstypen gibt es?	
3	Geben Sie eine Syntax für eine die jeweiligen Funktionen vor	
4	Was ist bei einer Rekursion wichtig zu beachten?	
5	Was ist der Unterschied zwischen einer lokalen und globalen Variablen?	
6	Was passiert bei Namensgleichheit der lokalen und globalen Variable?	
7	Was bedeutet Überladen von Funktionen?	
8	Was ist ein Default-Argument in Funktionen?	
9	Wie kann man Code bei C++ leicht trennen?	6.
10	Was sind Namensräume	6.3.3
11	Was sind Präprozessorbefehle und wo werden sie benutzt?	6.3.4

5.2 Beispiele

Nr.	Frage	Kapitel
1	Verarbeitung von Eingabewerten, um einen Ausgabewert zu produzieren. Steigerung der Lesbarkeit und Effizienz der Entwicklung, verbesserter modularer Aufbau	
2	Wiederkehrende Programmabschnitte, Übergabe von Werten, Rückgabe von Werten, Funktionen in Funktionen, Rekursionen	
3	<ol style="list-style-type: none"> 1. Void Fkt(); 2. Void Fkt (Argumentenliste) 3. Datentyp Fkt (Argumentenliste) { return Datentyp } 4. Void Fkt () { Fkt2() } 5. Void Fkt() { Fkt(); } 	
4	Abbruchbedingung.	
9	Cpp – Implementierungsdatei H – Deklarationsdatei / Schnittstellenbeschreibung	
10	Namensräume bei gleichlautenden Funktionen	
11	Sind vor der Bearbeitung des Compilers und gehören nicht zum Sprachumfang. Z.B, include	

5.3 Aufgaben

- a) Bilden Sie eine Summe in einer Funktion durch Addieren von 10 Zahlen und geben Sie diese in der Funktion noch aus.
- b) Bilden Sie eine Summe in einer Funktion durch Addieren von 10 Zahlen und geben Sie diese in der Main.cpp aus.
- c) Bilden Sie eine Mehrfachselektion in der Funktion ab (siehe dazu Bsp bei Mehrfachselektion)
- d) Schreiben Sie eine einfache Summenfunktion zu einer rekursiven Funktion um.

6 Arrays

6.1 Fragen

Nr.	Frage	Kapitel
1	Was ist Array?	7
2	Wie definiert man ein 1-dim Array,	7.1.1
3	Wie definiert man ein 2-dim Array,	7.1.2
4	Was ist der Unterschied bei einem mehrdimensionalen Array > 2 Dimensionen?	
5	Was ist ein Indexoperator?	
6	Was ist einer ForEach Schleife?	7.1.1
7	Was bedeutet int Werte = {0}	
8	Was bedeutet int Werte = {1,2,3,4}	
9	Was ist bei der Übergabe eines Arrays zu wissen?	
10	Wie kopiere ich ein Array?	
11	Was beschreibt eine Zeichenkette?	
12	Was bedeutet sortieren für ein Array? Was muss man beachten?	
13	Welche Sortiermethoden haben Sie gehört?	
14	Wie würden Sie einfach sortieren, wenn sie keine Sortiermethode kennen?	

6.2 Beispiele

Nr.	Frage	Kapitel
1	Array ist eine Zusammenfassung von mehreren Variablen, Sie werden über einen Indexoperator angesprochen. Sie werden auch Felder genannt.	
2	int werte[Anzahl Elemente], char werte[Anzahl Elemente]	
3	int werte[Anzahl Elemente] [Anzahl Elemente] entspricht Anzahlen Zeilen und Anzahl Spalten	
4	Indexoperator gibt die Stelle/Position im Array an, beginnend mit Index 0.	
5	Seit C++ 11 Standard kann durch eine Foreach Schleife in einem Array dies genutzt werden.	
6	Es werden alle Felder mit 0 vorbelegt.	
7	Arrays werden als Parameter übergeben und kann wie gewohnt zugegriffen werden z. fkt (int w[]); Es wird nicht das ganze Array übergeben, sondern die Anfangsadresse.	
8	Über einen Index für Index Kopieren	
9	Ein char Array mit einer Nullpointerterminierung z.b char name [5+1]	
10	Die Inhalte müssen getauscht werden	
11	Bubblesort, Sortieren nach Auswahl, Quicksort	
12	Über 2 Schleifen und mit dem ersten als Minimum setzten und durch die anderen schauen und dann entsprechend tauschen. Danach mit den 2. Usw.	

6.3 Aufgaben

- Schreiben Sie ein C++ Programm, das 5 Ganzzahlen über eine kopfgesteuerte Schleife einließen und danach wieder über eine fuß gesteuerte Schleife ausgibt. Nutzen sie eines eindimensionalen Arrays.

- b) Erweitern Sie ihr Programm, in dem sie beide Routinen als Funktion ausgliedern und das Array übergeben.
- c) Ersetzen sie die Schleifen durch eine ForEach Schleife.
- d) Schreiben Sie ein C++ Programm, das ein 8x8 Schachbrett abbildet. Dabei soll über der ersten Zeile die Buchstaben A – H erscheinen und vor jeder Zeile die Zeilen Nummer 1-8. A1 ist immer schwarz. Programmieren Sie eine Funktion, die ihn das die Farbe ausgibt, wenn Sie das Feld (z.B. A1) eingeben. z.B. Eingabe: A1: schwarz.
- e) Schreiben Sie ein C++ Programm, die ein Wort einlässt. Ermitteln die Wortlänge.
- f) Erweitern Sie b) in dem sie die Zahlen vorher sortieren nach einer einfach Sortiermethode
- g) Erweitern Sie b) mit Bubblesort
- h) Anspruchsvoll: Sortieren Sie eine beliebige Zeichenkette nach dem Alphabet und geben Sie die Buchstaben aus. Vergleichen Sie welche Buchstaben aus dem Alphabet fehlen und geben Sie diese aus.

7 Zeiger

7.1 Fragen

Nr..	Frage	Kapitel
1	Beschreiben Sie die Funktionsweise eines Zeigers	8
2	Wie geht die Deklaration eines Zeigers	
3	Was ist der Adressoperator?	
4	Was ist der Dereferenzierungsoperator	
5	Was bedeutet call by reference vs call by value?	8.2
6	Was ist der Unterschied zwischen Zeiger und Arrays?	
7	Was bedeutet Zeigerarithmetik?	
8	Kann ich Zeiger bei Funktionen machen?	
9	Was ist eine dynamische Speicherreservierung und wie erreicht man das?	
10	Was ist ein Referenzoperator?	8.3

7.2 Beispiele

Nr.	Frage	Kapitel
1	Zeigervariable ist eine Variable, die die Adresse einer anderen Variable speichert.	
2	<code>int * pi, double *pd</code>	
3	Die Adresse einer Variable erhält man über den Adressoperator <code>&</code> .	
4	Über den Dereferenzierungsoperator kann man auf den Inhalt der Adresse eines Zeigers zugreifen	
5	Call by reference = Übergabe der Adresse Call by value = Übergabe eines Wertes	
6	Zeiger und Arrays haben Gemeinsamkeiten, Array nichts anderes als ein reservierter Speicherbereich und Array Name steht für Anfangsadresse	
7	Es ist möglich Rechenoperation (Addition und Subtraktion) direkt auszuführen.	
8	Funktionen haben ebenso eine Startadresse im Speicher, daher auch als Zeiger aufrufbar.	
9	Erlaubt Reservierung des Speicherplatzes während der Laufzeit? Alles Array oder Variablen waren bisher statisch. Über <code>new</code> und <code>delete</code>	
10	Eine Referenz keine eigenständige Variable, Referenzen müssen bei der Deklaration initialisiert werden und über den <code>&</code> Operator gebildet.	

7.3 Aufgaben

- Definieren Sie dynamisch 10 Gleitkommazahlen und füllen Sie durch Eingabe über die Tastatur. Geben Sie diese Zahlen wieder aus.
- Nutzen Sie die Bsp. a) und erweitern Sie dies mittels Funktionen (d.h. Eingabe, Ausgabe, Definition Anzahl der Gleitkommazahlen)
- Erweitern Sie Bsp. b) mittels Sortierung. Nutzen Sie das Prinzip von Zeigern.

8 Strukturen

8.1 Fragen

Nr.	Frage	Kapitel
1	Was ist eine Struktur? Wie sieht die Deklaration aus?	9
2	Wie greift man auf die Inhalte einer Struktur?	
3	Können Strukturen mit Strukturen arbeiten?	
4	Können Arrays in Strukturen abgebildet werden?	9.1.4
5	Was sind Beispiele für höhere Datenstrukturen?	9.2
6	Was beschreibt eine einfache Liste?	
7	Welchen Unterschied hat eine doppelte Liste	
8	Was ist ein Binärbaum?	

8.2 Beispiele

Nr.	Frage	Kapitel
1	Eigener Datentyp und Zusammenfassung von Datentypen <code>struct TBruch { int Zaehler; int Nenner; };</code>	9
2	Über den Zugriff auf Operatoren (Punkt Operator)? Pfeiloperator	
3	Ja	
4	Zugriff erfolgt über Indexoperator. Dynamische Speicherreservierung funktioniert analog.	
5	Listen und Bäume	
6	Eine einfache Liste verweist immer auf das nachfolgende Element: <code>root->Element1->Element</code> . Hierbei wird das Prinzip der dynamischen Speicherreservierung genutzt. Es werden nur der Pointer/ Adresse des nächsten Elementes gespeichert	
7	Eine doppelte Liste hat zusätzliche die Information zum Vorgänger.	
8	Ein Binärbaum stellt 2 Nachfolger (2 Kinder, die für die nächste Ebene als Vater oder Mutter bekannt sind).	

8.3 Aufgaben

- Definieren Sie eine Struktur `Auto` aus folgenden Eigenschaften: `string name`, `int PS`, `int km_h`. Schreiben Sie eine C++ Programm und füllen Sie 2 Objekte mit „Audi, 204 PS und 250 km_h.“, „VW“, 150 PS, 200 km_h und geben Sie diese wieder aus.
- Definieren Sie eine einfache Liste
- Definieren Sie eine doppelte Liste

9 Klassen

9.1 Fragen

Nr.	Frage	Kapitel
1	Was ist ein Objekt?	10
2	Was sind Attribute und Methoden?	
3	Was ist eine Klasse?	
4	Wie sieht der generelle Aufbau einer Klasse aus? Erklären Sie die Struktur	10.1
5	Was ist ein Konstruktor?	
6	Was ist ein Destruktor?	
7	Was ist einer Parameterkonstruktor?	
8	Was sind sog. Get und Set- Methoden? Warum benötigen Sie diese?	10.1.4
9	Erklären Sie das Thema dynamische Speicherreservierung in Klassen	
10	Kann ich Zeichenketten einfach kopieren? Welchen Vorschlag haben Sie?	10.2.2
11	Was ist der sog. Genannte this Zeiger?	10.3.1
12	Wozu benötigt man statische Klassenelement. Geben Sie ein Beispiel	10.3.2
13	Worin liegt der Unterschied zwischen der Deklaration und Implementierung einer Klasse?	

9.2 Beispiele

Nr.	Frage	Kapitel
1	Eine softwaretechnische Repräsentation eines realen oder gedachten klar abgegrenzten Gegenstands	
2	Attribute sind Eigenschaften und beschreiben den Gegenstand. Methoden beschreiben die Operationen. Diese können von außen die Attribute beschreiben.	
3	Softwaretechnische Beschreibung eines Bauplanes für ein Objekt. Auf einer Klasse können Objekte abgeleitet werden-	
4	Class Name { Private: Attribute ->Datenkapselung Methoden Public: Attribute -> Schnittstelle Methoden };	
5	Für die direkte Initialisierung des eigenen Objektes. Es wird dadurch die Bildung eines Objektes gemacht.	
6	Methode wird aufgerufen, um die Gültigkeit des Objektes zu zerstören, d.h. aus dem Speicher zu löschen.	
7	Der Standardkonstruktor wird um einer Argumentenliste (Attribute der Klasse erweitert)	
8	Im Wege der Datenkapselung müssen die Attribute durch set und Get Methoden öffentlich gemacht werden (Schnittstellengedanke)	
9	Anlog der dyn. Speicherreservierung nur in einer Klasse. Nachträgliche Definition des Speichers	
10	Nein, strcpy ist notwendig.	

11	Ist ein Zeiger, der beim Aufruf einer Methode eines Objektes automatisch mitgeliefert wird und zeigt auf das Objekt, welche die Methode aufrufen will. (sog. Selbstbezug)	
12	Man erhält separaten Speicherplatz für seine Attribute, dadurch könne beliebig viele bestehen. Es macht sinn manchmal nur alle Objekte ein Attribut zu nehmen (sog. Klassenattribut). Instanziierung erfolgt außerhalb der Klasse.	
13	Im Sinne der Modularisierung werden Deklaration in Header und Implementierung in CPP Datei gespeichert. Die Namensgebung orientiert sich nach dem Klassennamen.	

9.3 Aufgaben

- Bilden Sie eine Klasse Schueler bestehend aus name, vorname, geburtstag.
- Bilden Sie einen Konstruktor und einen Parameterkonstruktor
- Bilden Sie einen Destruktor
- Schreiben Sie ein C++ Programm das 2 Objekte Person generiert:
 - Person (Name1, Vorname1, 10.10.22)
 - Person (Name2, Vorname2, 20.20.22)geben Sie beide Objekte wieder aus.
- Bilden Sie die entsprechenden Set-/Get Methoden
- Zeigen Sie ein Beispiel mit dynamische Speicherreservierung und Klasse.
- Trennen Sie das Beispiel in eine Deklaration und eine Implementierungsdatei

10 Dateioperationen

10.1 Fragen

Nr.	Frage	Kapitel

10.2 Beispiele

Nr.	Frage	Kapitel

10.3 Aufgaben

a) ..

11 Vererbung / Polymorphie

11.1 Fragen

Nr.	Frage	Kapitel
1	Welche Arten von Vererbung gibt es?	13 / 14
2	Wie funktioniert die Einfache Vererbung?	13.1
3	Wie sieht die Deklaration für eine einfache Vererbung aus?	13.1.1
4	Welche Bedeutung hat das Schlüssel protected bei der einfachen Vererbung?	13.1.2
5	Was bedeutet der Aufruf eines Basisklassenkonstruktors?	13.1.3
6	Wie funktioniert die mehrfache Vererbung?	13.2
7	Wie sieht die Deklaration für eine mehrfach Vererbung aus?	
8	Was bedeutet virtuelle Vererbung?	13.2.2
9	Wie sieht der Basisklassenkonstruktor aus?	13.2.3
10	Was bedeutet Polymorphismus?	14
11	Was sind virtuelle Methoden	14.2.1
12	Was ist eine abstrakte Klasse und wozu benötigt diese man?	14.2.4

11.2 Beispiele

Nr.	Frage	Kapitel
1	Einfache Vererbung, mehrfach Vererbung, Polymorphie	
2	Bei der einfachen Vererbung wird spezialisiert, d.h. abhängig von der Oberklasse	
3	Class Vererbte Klasse: public Oberklasse. Public/private/protected steht hier für Vererbung	
4	Es ermöglicht direkt auf die Attribute der Oberklasse zuzugreifen, anstatt über die public Methode zu gehen (Prinzip Datenkapselung)	
5	Die vererbte Klasse hat alle Funktionalitäten der Oberklasse. Es gibt Methoden, die nach der Instanziierung direkt aufgerufen werden und danach nicht mehr aufrufbar sind (z.B. Konstruktor). Sie müssen neu installiert werden durch eine zusätzliche Addition der Oberklasse (Parameter)	
6	Bei der Mehrfachvererbung werden von 2 Klassen vererbt.	
7	Class Klasse: public Oberklasse 1, public Oberklasse 2. Hinweise: Anpassen des Basisklassenkonstruktor.	
8	Bei der virtuellen Vererbung versucht man zu verhindern, dass zu einer doppelten Vererbung der Oberklasse kommt,	
9	Es müssen die Basisklassenkonstruktor explizit die Konstruktoren der Oberklassen aufrufen. Beginnen mit der ersten Klasse. Ich kann auch durch einen Parameterkonstruktor erfolgen.	
10	Polymorphismus bedeutet, dass ein Zeiger während der Laufzeit auf verschiedene Objekte zeigen kann und dann trotzdem die korrekten Methoden aufrufen kann, die über den Zeiger aufrufbar sind. Wichtig ist ein Zeiger auf die Basisklasse	
11	Virtuelle Methoden lösen bei Polymorphismus das Problem der korrekten Nutzung von Methoden	
12	Ist eine Klasse, von der keine Objekte instanziiert werden können. Sie dient als Grundlage für weitere Klassen einer Vererbungshierarchie. Sie wird durch min eine virtuelle Methode gekennzeichnet.	

	Virtuale Rückgabetyt Name (Parameter) = 0;	
--	--	--

11.3 Aufgaben

a) ..

12 PAP

12.1 Fragen

Nr.	Frage	Kapitel
1	Was ist ein Programmablaufplan?	
2	Wie sieht die Notation aus?	

12.2 Beispiele

Nr.	Frage	Kapitel

12.3 Aufgaben

- a) Bilden Sie eine kopfgesteuerte Schleife
- b) Bilden Sie eine zählgesteuerte Schleife
- c) Bilden Sie eine fussgesteuerte Schleife

13 Struktogramm

13.1 Fragen

Nr.	Frage	Kapitel
1	Was ist ein Struktogramm?	
2	Wie sieht die Notation aus?	

13.2 Beispiele

Nr.	Frage	Kapitel

13.3 Aufgaben

- a) Bilden Sie eine kopfgesteuerte Schleife
- b) Bilden Sie eine zählgesteuerte Schleife
- c) Bilden Sie eine fuß gesteuerte Schleife

14 Klassendiagramm

14.1 Fragen

Nr.	Frage	Kapitel
1	Was ist ein Klassendiagramm?	
2	Welche Notation Sprache kennen Sie für einen Klassendiagramm?	
3	Was bedeutet Aggregation?	
4	Was bedeutet Komposition?	
5	Was sind Kardinalitäten in einem Klassendiagramm?	
6	Was bedeuten die Notation +, -, # in einem Klassendiagramm?	

14.2 Beispiele

Nr.	Frage	Kapitel
1	Darstellung von Klassen in einer grafischen Darstellung	
2	UML	
3		
4		
5		
6	Public, private, protected	

14.3 Aufgaben

- Definieren sie ein einfache Klassendiagramm aus der Klasse Person, Lehrer, Schüler. Nutzen Sie eine einfache Vererbung
- Definieren sie ein einfache Klassendiagramm aus der Klasse Person und Fach, Lehrer und Schüler. Nutzen Sie eine mehrfache Vererbung. Erklären Sie ihre Lösung.
- Welche Methoden würden Sie in Beispiel a) protected, private oder public machen. Erklären Sie warum?
- Welche Kardinalität würden Sie in ihrem Klassendiagramm für a) einführen
- Überlegen Sie nach c) für Beispiel b)
- Welche Kardinalitäten würden Sie in Beispiel b) einführen