



MINISTERIUM FÜR KULTUS, JUGEND UND SPORT  
BADEN-WÜRTTEMBERG

ABITURPRÜFUNG AM BERUFLICHEN GYMNASIUM IM SCHULJAHR 2012/2013

Hauptprüfung	LÖSUNGSVORSCHLAG FÜR DAS FACH
1.5.2	Informationstechnik (TG)

Arbeitszeit	270 Minuten	
Hilfsmittel	eingeführte Formelsammlung und lokale Formelsammlung zugelassener Taschenrechner	
Stoffgebiet	<p>Teil 1: Hardware Aufg. 1: Informationsverarbeitende Systeme (2 Seiten) Aufg. 2: Informationsverarbeitende Systeme (5 Seiten)</p> <p>Teil 2: Software Aufg. 3: Objektorientierte Analyse und Design (3 Seiten) Aufg. 4: Objektorientierte Analyse und Design (4 Seiten)</p> <p>Teil 3: Systeme Aufg. 5: Datenbank- und Betriebssysteme/vernetzte Systeme (3 Seiten) Aufg. 6: Datenbank- und Betriebssysteme/vernetzte Systeme (3 Seiten)</p> <p>Übersicht: Punkteverteilung nach Anforderungsbereiche (1 Seite)</p>	
Bemerkungen	<p><b>LÖSUNGSVORSCHLAG</b></p> <p><b>nur für die Fachlehrerin/den Fachlehrer bestimmt</b></p>	

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)
1.5.2	Informationstechnik (Hardware)
Lösungsvorschlag	Teil: 1 (Pflichtbereich)      Aufgabe 1: ( 2 Seiten)

Punkte

1.1.1	Start	A	B	Takt
1	1	1		0
0	x	x		1
x	0	x		1
x	x	0		1

1.1.2

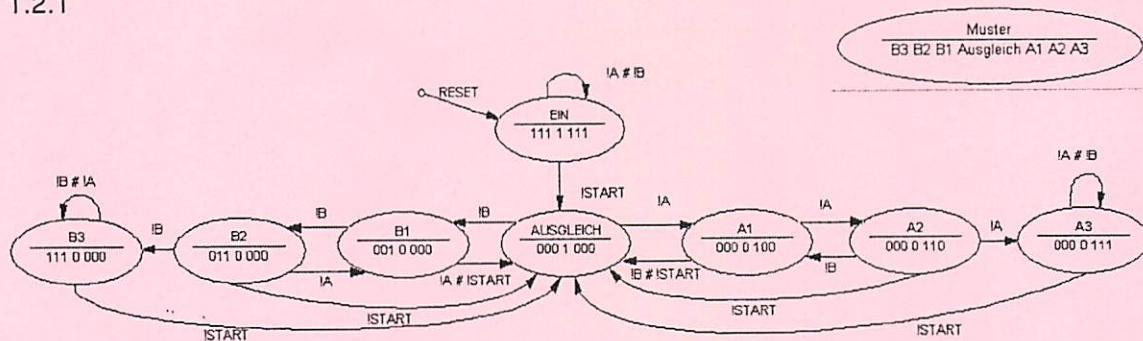
2  
1

$$\text{Takt} = \text{!Start} \# \text{!A} \# \text{!B}$$

Je nach Tabelle zu 1.1.1 wäre auch eine EXOR-Verknüpfung korrekt.

1.2.1

5



Muster  
B3 B2 B1 Ausgleich A1 A2 A3

1.2.2 8 Zustände  $\Rightarrow 2^3 = 8 \Rightarrow 3 \text{ FF}$

1

1.2.3

2

Zustand	Q2	Q1	Q0
EIN	0	0	0
AUSGLEICH	0	0	1
A1	0	1	0
A2	0	1	1
A3	1	0	0
B1	1	0	1
B2	1	1	0
B3	1	1	1

Start	A	B	n			n+1		
			Q2	Q1	Q0	Q2	Q1	Q0
0	1	1	x	x	x	0	0	1
1	0	1	0	0	1	0	1	0
1	1	0	0	1	0	0	0	1

1.2.4 3 Eingangssignale und 3 Zustandssignale  $\Rightarrow$  maximal  $2^6 = 64$  Zeilen oder entsprechend der benötigte Übergänge hier 18, wobei 5 davon doppelt sind (ODER-Verknüpfung)  $\Rightarrow$  23 Zeilen notwendig.

2

1.3.1 Grundlage:  
Takt = 12MHz  
 $\Rightarrow 1 \text{ Maschinenzyklus} = \text{Takt}/12$   
 $\Rightarrow 1 \text{ MZ} = 1\mu\text{s}$   
 $3\text{s} = 3.000.000\mu\text{s}$   
 $\Rightarrow 1.500.000 \times 2\mu\text{s} (\text{DJNZ})$   
 $\Rightarrow \text{Startwerte} = 150 \times 100 \times 100$

wait_3s:	push R7	; Registerwerte
	push R6	; auf Stack retten
	push R5	
	MOV R7,#100	; Registerwerte setzen
M0:	MOV R6,#100	; $100 * 100 * 150 * 2$
M1:	MOV R5,#150	; $= 3.000.000 \rightarrow 3 \text{ Sek.}$
M2:	DJNZ R5,M2	; $150 * 2$
	DJNZ R6,M1	; $* 100$
	DJNZ R7,M0	; $* 100 = 3.000.000$
	pop R5	; alte Registerwerte
	pop R6	; vom Stack laden
	pop R7	
RET		; Rücksprung aus UP

Anmerkung zum Quellcode:  
Es obliegt dem Lehrer in wie weit er die push und pop-Befehle in die Bewertung aufnimmt.

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Hardware)	
Lösungsvorschlag	Teil: 1 (Pflichtbereich)	Aufgabe 1: ( 2 Seiten)

Punkte

1.3.2 Es ist nicht eindeutig, wie ein Compiler den Quellcode der höheren Programmiersprache in die Maschinensprache übersetzt. Somit kann nicht auf eine bestimmte Abfolge der Maschinenbefehle zurückgegriffen werden. 1

1.3.3 Mittels Timer, da hierbei unabhängig von Befehlszeiten die Maschinenzyklen gezählt werden. 1

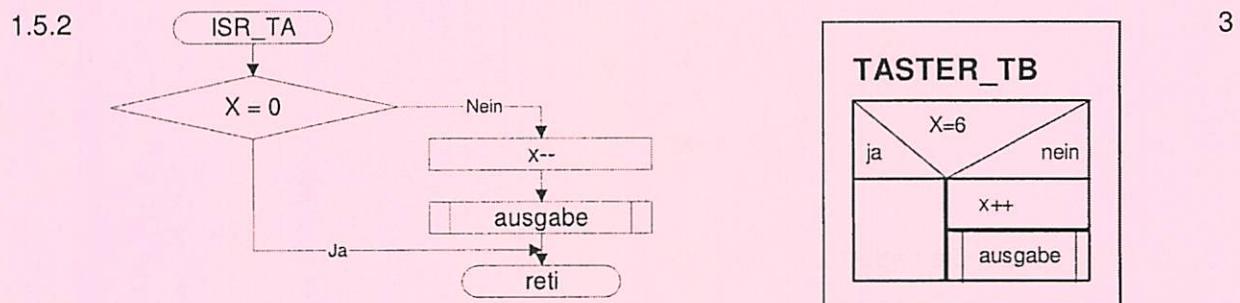
1.4 punkte: db 0x07 , 0x06 , 0x04 , 0x08 , 0x10 , 0x30 , 0x70  
 ausgabe: mov dptr,#punkte ; Datenpointer auf Tabellenanfang  
 mov A,R7 ; Referenzwert X (R7) in Akku  
 movc A,@A+dptr ; Ausgabekodierung für X (R7) laden  
 mov N,A ; und auf dem Port N ausgeben  
 ret

---

```
char code punkte[7] = { 0x07 , 0x06 , 0x04 , 0x08 , 0x10 , 0x30 , 0x70 };
void ausgabe(char x) // Funktionskopf; Übergabeparameter kann auch
{
    N = punkte[x]; // globale Variable sein
    // Feldelement x auf Port N ausgeben
}
```

1.5.1 TA an P3.2 für externen Interrupt 0 und TB an P3.3 für externen Interrupt 1 3

```
init: setb IT0 ; Flankensteuerung externer Interrupt 0
      setb EX0 ; Freigabe der ISR zum externen Interrupt 0
      setb IT1 ; Flankensteuerung externer Interrupt 1
      setb EX1 ; Freigabe der ISR zum externen Interrupt 1
      setb EAL ; allgemeine ISR-Freigabe
```

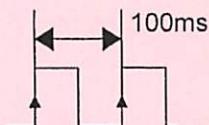


1.5.3 Jedem Interrupt (extern oder intern) ist hardwaremäßig eine unveränderbare Vektoradresse zugeordnet. Mit dieser Adresse wird der Programmcounter (PC) bei Auftreten des Interrupts geladen, so dass der Programmcode ausgeführt wird, der an dieser Adresse steht. Im Gegensatz hierzu kann ein Unterprogramm an einer beliebigen Adresse stehen, da diese beim Aufruf angegeben wird. 1

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)
1.5.2	Informationstechnik (Hardware)
Lösungsvorschlag	Teil: 1 (Pflichtbereich)      Aufgabe: 2 (5 Seiten)

Punkte

2.1.1



$$f = 1/T = 1/100\text{ms} = 10\text{Hz}$$

1

2.1.2 Aus der Zustandsübergangstabelle lässt sich erkennen, dass das Schaltwerk 6 (sinnvolle Zustände unterscheidet. Aufgrund der 3 Speicher sollte die Antwort 8 ( $2^3$ ) aber auch akzeptiert werden.

1

2.1.3

$Q_2$	$Q_1$	$Q_0$	laden
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1

$$\neg \text{laden} = (\neg Q_2 \wedge \neg Q_1 \wedge \neg Q_0)$$

$$\text{laden} = (\neg Q_2 \wedge \neg Q_1 \wedge \neg Q_0)$$

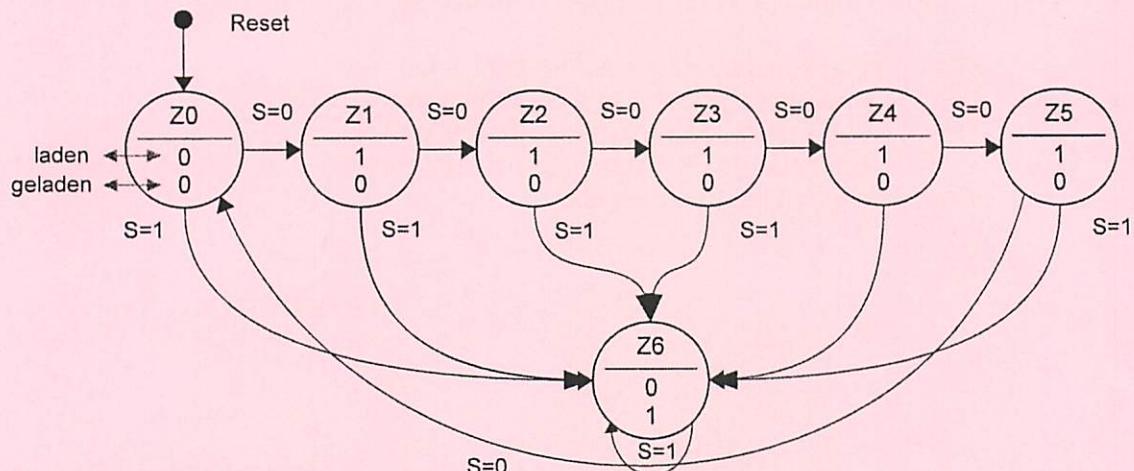
$$\text{laden} = Q_2 \# Q_1 \# Q_0$$

... oder andere Lösungen (diese Codierung ist nicht vorgegeben!)

3

2.1.4 Zustandsdiagramm:

5



▷ in diesem Fall liegt die Ladepause am Anfang ... dies ist aber nicht vorgegeben!

Zustandsübergangstabelle  
(nicht verlangt)

S	n			n+1		
	$Q_2$	$Q_1$	$Q_0$	$Q_2$	$Q_1$	$Q_0$
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	1	1
0	0	1	1	1	0	0
0	1	0	0	1	0	1
0	1	0	1	0	0	0
0	1	1	0	0	0	0
0	1	1	1	1	1	1
1	X	X	X	1	1	1

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Hardware)	
Lösungsvorschlag	Teil: 1 (Pflichtbereich)	Aufgabe: 2 (5 Seiten)

Punkte

2.1.5 2

Frequenz verdoppeln (von 10Hz auf 20Hz  $\Rightarrow$  50ms Periodendauer)

1s : 0,05s entspricht einem Verhältnis von 20:1

$\diamond$  man muss 21 Zustände unterscheiden.

Mit 5 Speichern erreicht man  $2^5 = 32$  unterschiedliche Zustandscodierungen

$\diamond$  5 Speicher reichen aus.

Wenn man berücksichtigt, dass für den „geladen“ - Zustand ein weiterer Zustand notwendig wird (s. 1.4), sind sogar 22 Zustände notwendig.

2.2.1 mit einem 16-Bit-Regitser erreicht man bei einer Zeitbasis von  $1\mu s \cdot 2^{16} = 65536 \mu s = 0,065536s$ . 3

$\diamond$  Um auf längere Zeiten zu kommen, sind zusätzliche „Zählregister“ notwendig.

$\diamond$  man muss den Zählbereich des Timers so setzen, dass mit ganzzahligen Vielfachen der gewünschte Zeitraum erreicht werden kann.

$\diamond$  für 5 Sekunden könnte man 50ms (16-Bit-Timer) x 100mal ausführen. Mit einem 8-Bit-Timer (0,25ms x 20000) wären schon zwei Zählregister notwendig.

... oder sinngemäß

2.2.2 2

z.B. mit Timer T0 beim MC8051

```
JMP init
; -----T0 ISR-----;
; ;
init: MOV TMOD,#0000 0001b ; 16-Bit-Timer
      MOV TL0,#0B0h ; Lowbyte von T0
      MOV TH0,#3Ch ; Highbyte von T0
                  ; T0 zählt von 0 ... 65536 und löst dann einen
                  ; Interrupt aus  $\diamond$  wenn man bei 15536 startet,
sind
                  ; es 50000 Schritte ( $\mu s$ ) pro Interrupt
                  ; 15536dezimal = 3CB0hex

SETB EA ; Interruptfreigabe (allgemein)
SETB ET0 ; T0 - Interruptfreigabe
SETB TR0 ; T0 starten
```

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Hardware)	
Lösungsvorschlag	Teil: 1 (Pflichtbereich)	Aufgabe: 2 (5 Seiten)

## Punkte

2.2.3 2

ORG 0B0h CLR TR0	; T0 stoppen
MOV TL0,#0B0h	; T0 zählt von 0 ... 65536 und löst dann einen
MOV TH0,#03Ch	; Interrupt aus $\Rightarrow$ wenn man bei 15536 startet, sind ; es 50000 Schritte ( $\mu$ s) pro Interrupt
SETB TR0	; T0 wieder starten
SETB laden	; nach 50ms wird geladen
DJNZ R7, weiter	; wenn man z.B. mit R7 20 Zählschritte realisiert, ; hat man genau 1 Sekunde erreicht
CLR laden	; Ladepause nach 20 x 50ms
MOV R7,#21	; Ladezeit wieder initialisieren

weiter: RETI

2.2.4 2

- der Interrupt wird durch das Timerüberlaufflag ( $TF0=1$ ) ausgelöst und durch RETI wieder gelöscht.
- die **aktuelle Programmadresse** wird vom Programmcounter (PC) auf dem **Stack** gespeichert. Die Einsprungadresse des zugehörigen Interrupts ( $T0 \Rightarrow 00B0h$ ) wird in den PC geladen
- das Programm wird an der Interrupeinsprungadresse fortgesetzt.
- die ISR (Interruptsubroutine) wird abgearbeitet bis zum Befehl <RETI>
- die obersten **beiden Byte** des Stack (= Programmadresse, wo das Programm vorher unterbrochen wurde) werden **vom Stack** in den Programmcounter (PC) geholt
- das Programm wird fortgesetzt

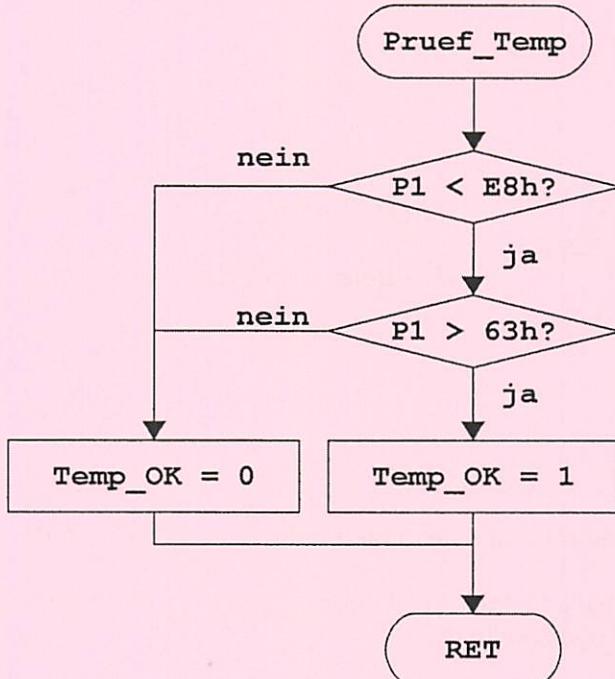
... oder sinngemäß

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)
1.5.2	Informationstechnik (Hardware)
Lösungsvorschlag	Teil: 1 (Pflichtbereich)      Aufgabe: 2 (5 Seiten)

Punkte

2.3

3



2.4.1

U\_speichern:

```

MOV AD10,AD9      ;ADn sind Speicher im internen RAM zugewiesen
MOV AD9,AD8
MOV AD8,AD7
MOV AD7,AD6
MOV AD6,AD5
MOV AD5,AD4
MOV AD4,AD3
MOV AD3,AD2
MOV AD2,AD1
MOV AD1,A
RET
  
```

Varianate mit Schleife:

U\_speichern:

```

MOV R0,#88h      ; indirekt adressierbarer Bereich
weiter:          MOV R7,@R0      ; Inhalt von 80 nach R7 (R7 als Zwischenspeicher)
                  DEC R0       ; nächst höhere Adresse im RAM
                  MOV @R0,R7      ; R7 an diese höhere Adresse kopieren
                  CJNE R0,#80h,weiter ; nach 9 Verschiebungen
                  MOV @R0,P1      ; aktuellen Wert an Adresse 80h
                  RET
  
```

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Hardware)	
Lösungsvorschlag	Teil: 1 (Pflichtbereich)	Aufgabe: 2 (5 Seiten)

Punkte

4

Variante in „C“:

```
U_speicher ()
{
    for (x=0; x<=8 ; x++)
        AD[x+1] = AD[x];           // AD2 ... AD10
    AD[0]=P1;                   // aktuellster Messwert → AD1
}
```

2.4.2

2

Die Spannung steigt während des gesamten Ladeverlaufs bis zum Punkt B an.

Vom Punkt B zum Punkt C fällt der Wert der Spannung ab. Im Datenspeicher werden immer zehn Messwerte gespeichert.

So kann man den aktuellen Wert mit dem vorhergegangenen vergleichen und wenn der Messwert am Port kleiner als der Messwert AD1 ist den Ladevorgang beenden.

—

30

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Software)	
Lösungsvorschlag	Teil: 2 (Pflichtbereich)	Aufgabe: 3 (3 Seiten)

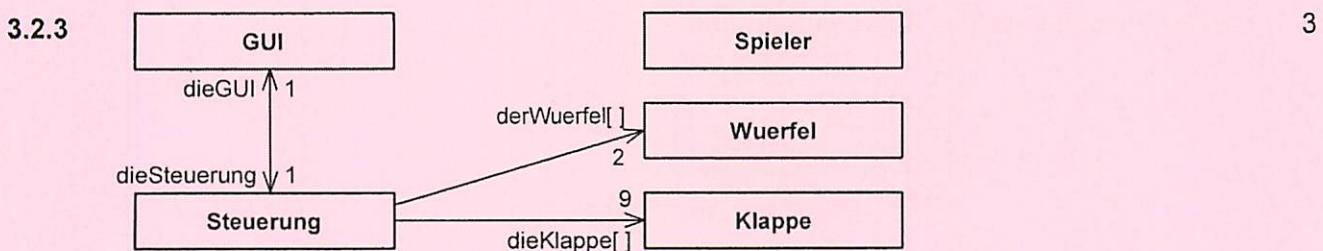
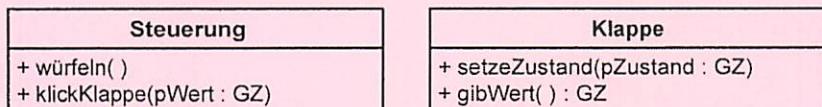
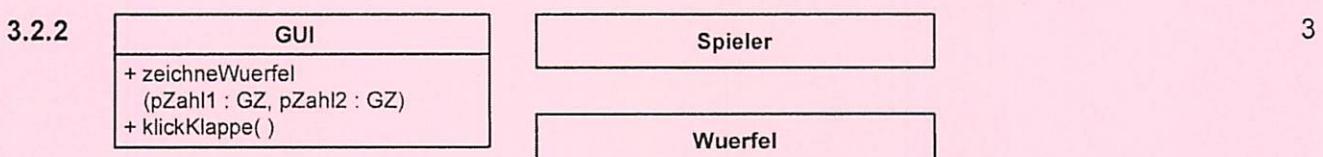
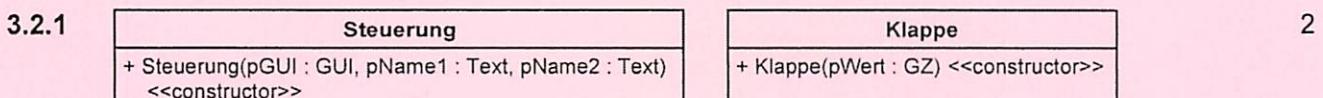
Punkte

### Aufgabe 3 Computerspiel 'Shut the Box'

#### 3.1 Analyse des Klassendiagramms

- 3.1.1 Der Konstruktor der Spielerklasse besitzt den Parameter *pName : Text*. Bei der Erzeugung eines Spielerobjekts muss der Name des Spielers übergeben werden. Innerhalb des Konstruktors wird der Parameter *pName* dem Attribut *aName : Text* zugewiesen. Da die Klasse keine Operation zum Schreiben des Attributs *aName* besitzt und das Attribut *aName* privat deklariert ist, kann der Wert des Attributs *aName* nicht mehr geändert werden. 2
- 3.1.2 Die Klasse *Spieler* besitzt das privat deklarierte Attribut *aMinusPunkte : GZ*. Operationen der Klasse können auf das Attribut zugreifen. Von außerhalb der Klasse kann nur über die öffentlich deklarierten Schreib- und Leseoperationen *setzeMinusPunkte( )* und *gibMinusPunkte( ) : GZ* auf das Attribut zugegriffen werden. Dieser Zugriff auf privat deklarierte Attribute über öffentlich deklarierte Schreib- und Leseoperationen heißt Datenkapselung. 1

#### 3.2 Klassendiagramm ergänzen



#### Begründung der Richtungen und der Kardinalität:

*GUI ↔ Steuerung*

Das Oberflächenobjekt sendet Botschaften an das Steuerungsobjekt (*wuerfeln( )*), das Steuerungsobjekt sendet ebenfalls Botschaften an das Oberflächenobjekt (*zeichneWuerfel( )*). Damit ist die Assoziation bidirektional. Die Kardinalität ist jeweils 1, da nur ein Steuerungs- und ein Oberflächenobjekt vorhanden ist.

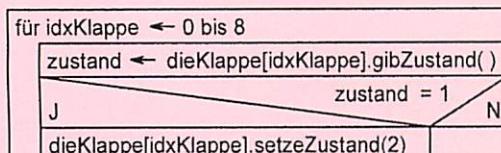
*Steuerung → Wuerfel*  
*Steuerung → Klappe*

Das Steuerungsobjekt sendet Botschaften an die Würfel- (*wuerfeln( )*) und an die Klappenobjekte(*setzeZustand( )*), jedoch nicht umgekehrt. Unidirektionale Assoziationen mit der Richtung *Steuerung → Wuerfel* bzw. *Steuerung → Klappe* sind ausreichend. Es sind zwei Würfelobjekte (Kardinalität = 2) und neun Klappenobjekte (Kardinalität = 9) vorhanden.

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Software)	
Lösungsvorschlag	Teil: 2 (Pflichtbereich)	Aufgabe: 3 (3 Seiten)

Punkte

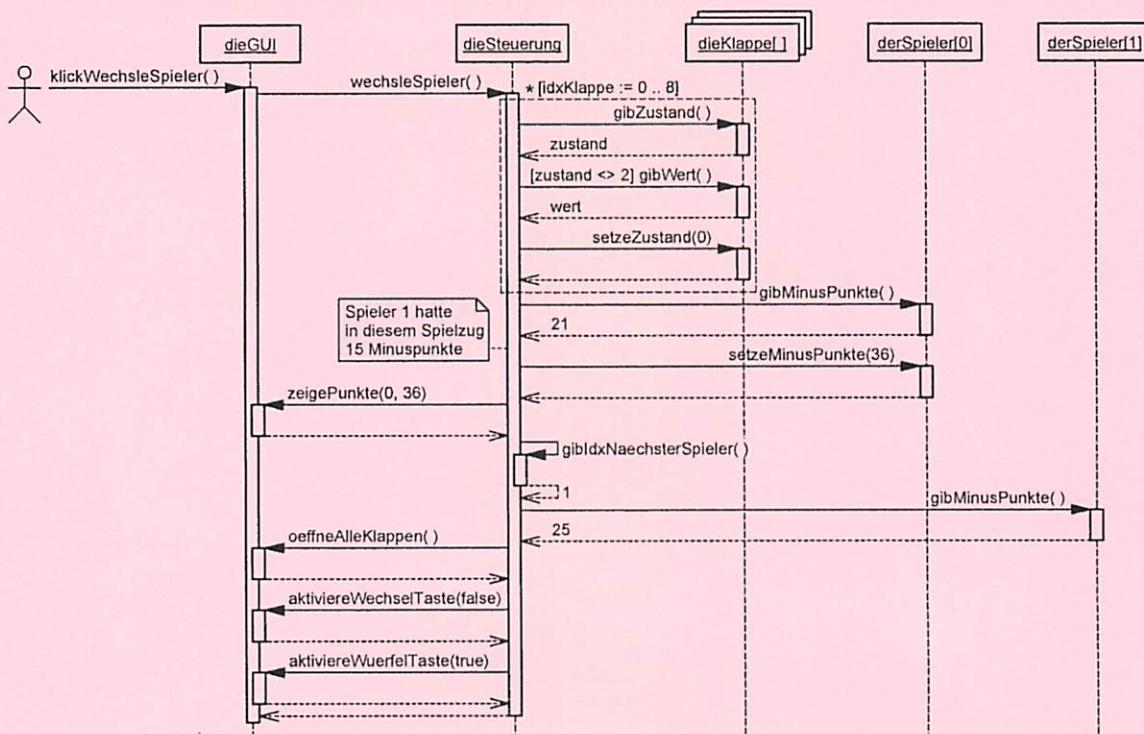
### 3.3 Struktogramm aus Sequenzdiagramm ableiten



2

### 3.4 Sequenzdiagramm aus Struktogramm ableiten

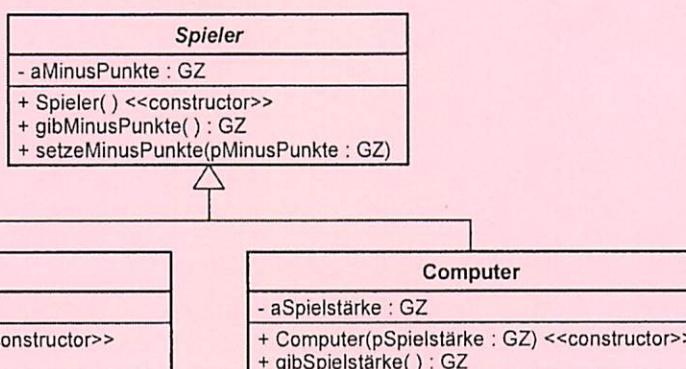
5



### 3.5 Weiterentwicklung des Klassendiagramms

#### 3.5.1

3



Hinweis: Die Klasse *Spieler* könnte auch konkret deklariert sein.

#### 3.5.2 Variante 1: Abstrakte Superklasse

2

Eine Instanz der Superklasse *Spieler* macht keinen Sinn, da diese unvollständig wäre. Durch die abstrakte Deklaration der Klasse *Spieler* wird verhindert, dass ein Objekt der Klasse *Spieler* erzeugt werden kann.

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Software)	
Lösungsvorschlag	Teil: 2 (Pflichtbereich)	Aufgabe: 3 (3 Seiten)

Punkte

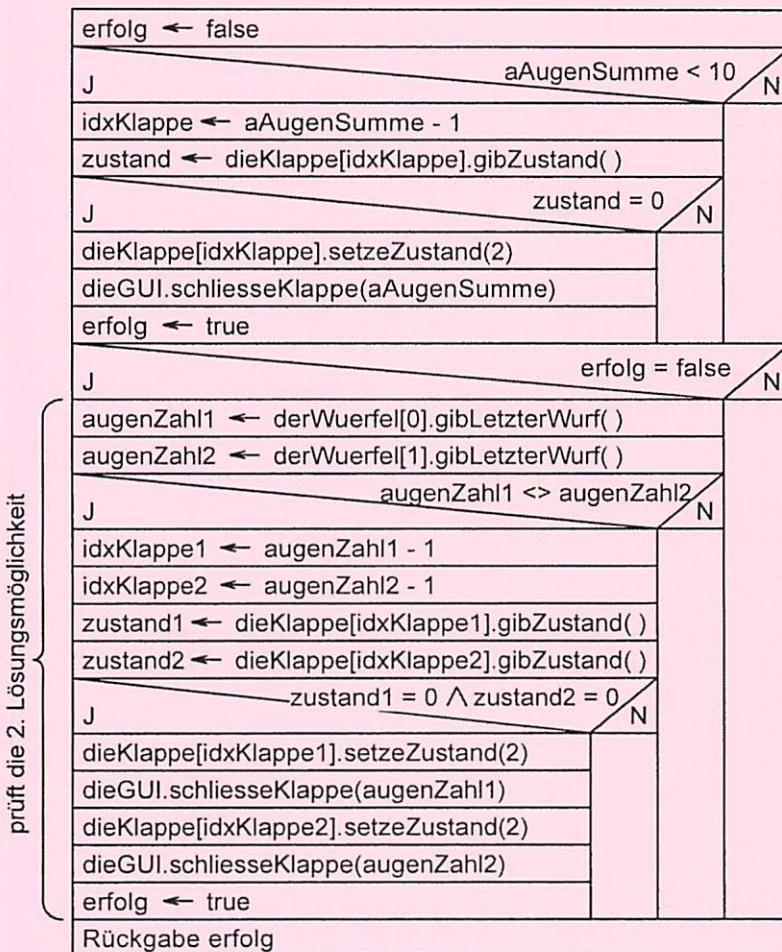
**Variante 2:** Konkrete Superklasse

Die beiden Subklassen *Mensch* und *Computer* besitzen keine Operationen, die sich als gemeinsame abstrakte Operation für die Superklasse *Spieler* eignet.

Wegen den unterschiedlichen Rückgabeparametern sind die Lese-Operationen *gibName()* : *Text* und *gibSpielstärke()* : *GZ* zu unterschiedlich, so dass sich ein Überschreiben einer abstrakten Operation der Superklasse in den Subklassen nicht anbietet. Eine abstrakte Klasse ist hier nicht notwendig.

## 3.6 Struktogramm entwickeln

7

*computerZug()* : Boolean

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Software)	
Lösungsvorschlag	Teil: 2 (Pflichtbereich)	Aufgabe: 4 (4 Seiten)

Punkte

#### 4.1 Grundlagen der Objektorientierung

4.1.1

Spieler: Mario

1

Trainer: Horst

Person: Hans

Spieler: Lukas

Mitarbeiter: Sepp      Leerzeichen und Zeilenumbrüche sind nicht bewertungsrelevant

4.1.2

Dynamische Polymorphie, spätes Binden: Objekte von Unterklassen können z.B. in einem Array der Oberklasse verwaltet werden. Wird eine in der Oberklasse deklarierte Operation auf ein Element des Arrays aufgerufen, wird tatsächlich die in der Unterkasse überschriebene Operation ausgeführt. Es wird erst zur Laufzeit entschieden, welche Operation ausgeführt wird.

2

4.1.3

Es kann beim Compilieren nicht garantiert werden, dass die Botschaft an alle Elemente des Arrays gesandt werden kann, da diese Operation nicht in der Oberklasse deklariert wurde. Deshalb würde dieser Ausdruck nicht compiliert werden.

2

4.1.4

Das Objektdiagramm würde anders aussehen. Es könnte kein Objekt der Klasse *Person* existieren, da von abstrakten Klassen keine Objekte erzeugt werden können.

1

4.1.5

Es wäre sinnvoll, die Klasse Person abstrakt zu deklarieren, da diese Klasse eine Verallgemeinerung der anderen konkreten Klassen ist. Allen anderen lässt sich eine konkrete Aufgabe oder Funktion zuordnen.

2

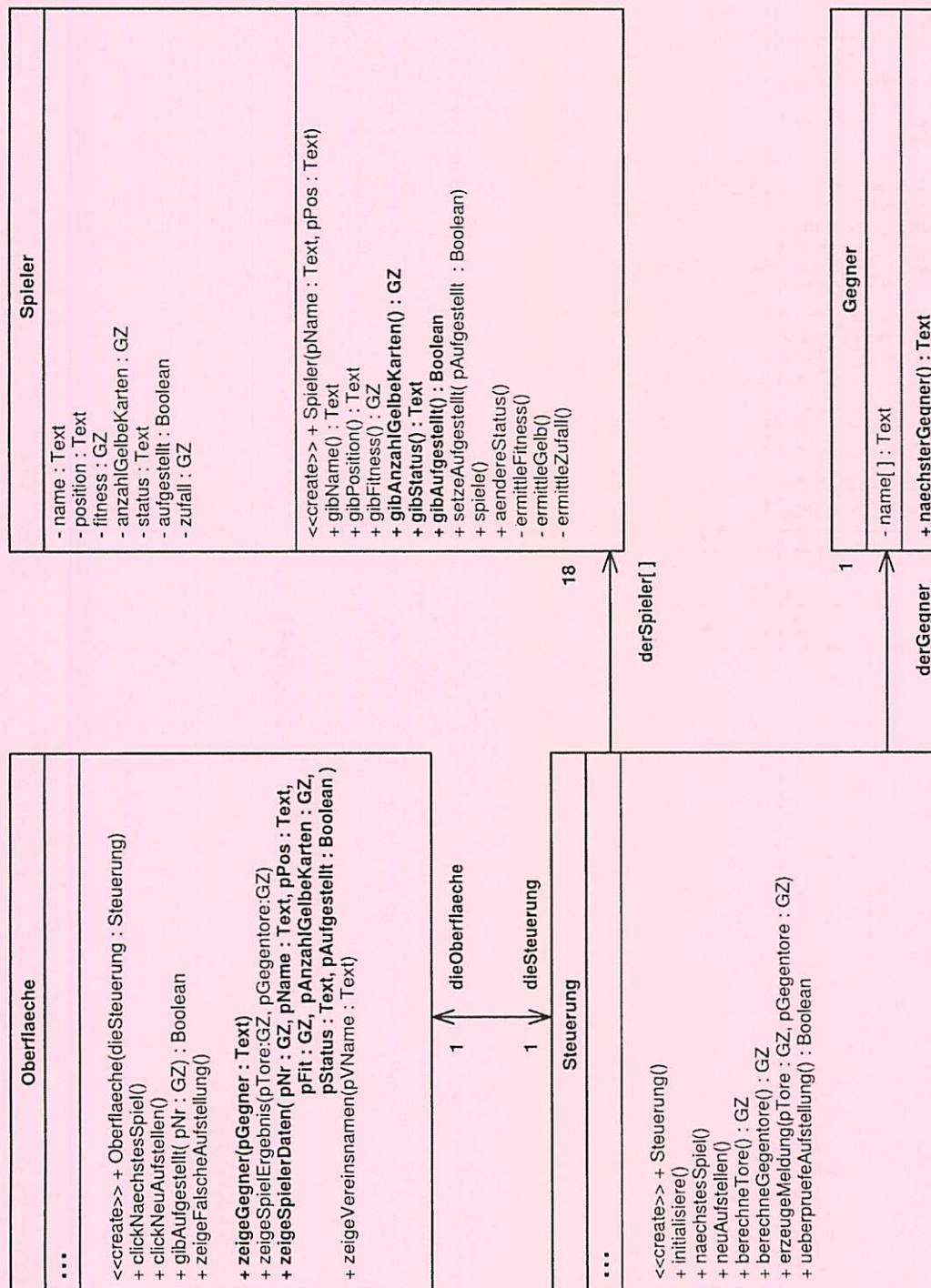
Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Software)	
Lösungsvorschlag	Teil: 2 (Pflichtbereich)	Aufgabe: 4 (4 Seiten)

Punkte

4.2.1 Klassendiagramm anhand eines Sequenzdiagrammes weiterentwickeln  
(4.2.1.1+4.2.1.2)

4

3



Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Software)	
Lösungsvorschlag	Teil: 2 (Pflichtbereich)	Aufgabe: 4 (4 Seiten)

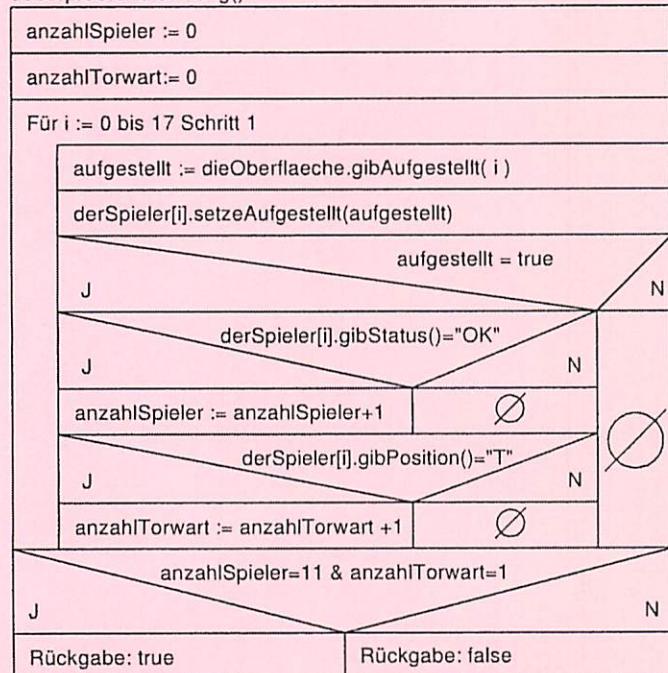
Punkte

#### 4.2.2 Algorithmus entwickeln

Mögliche Lösung:

8

ueberpruefeAufstellung()



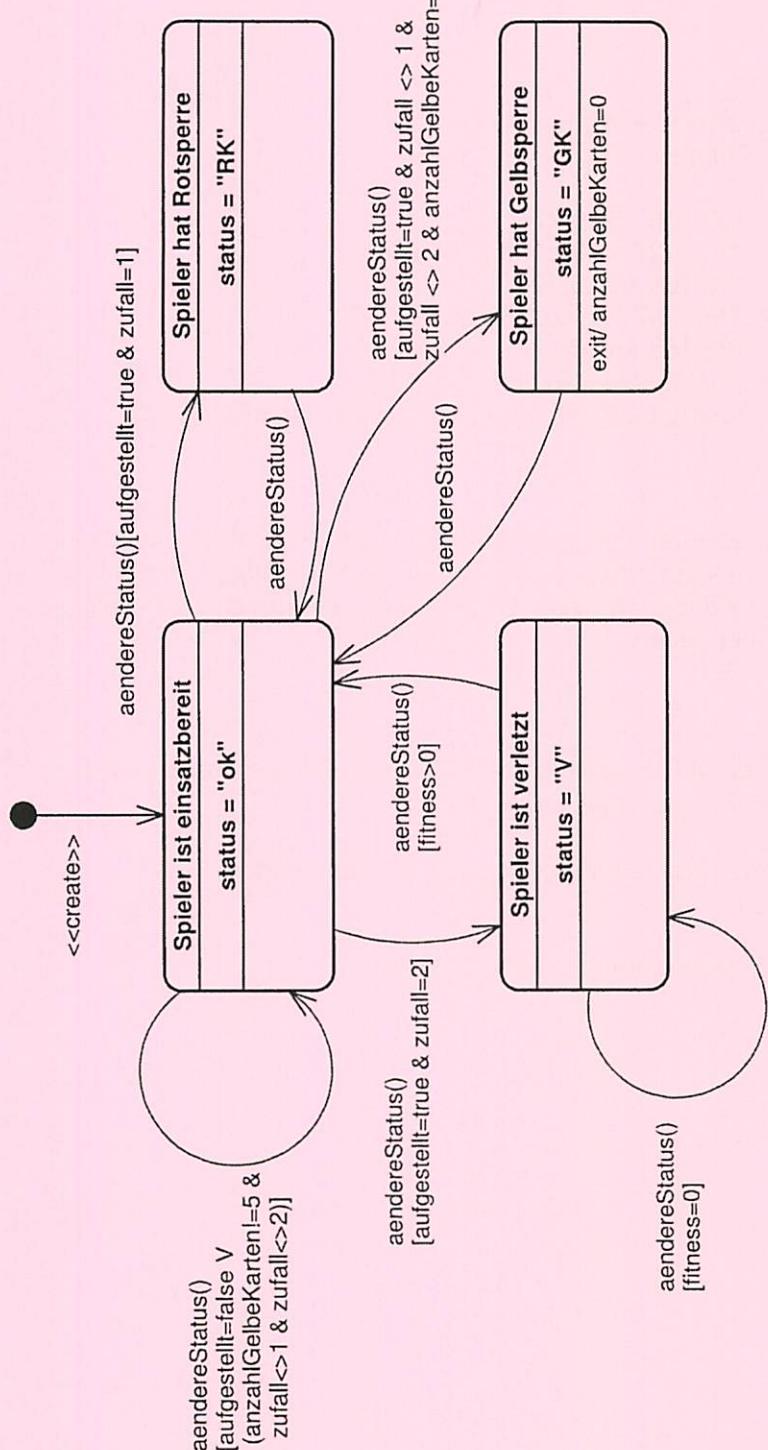
Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)
1.5.2	Informationstechnik (Software)
Lösungsvorschlag	Teil: 2 (Pflichtbereich)

Aufgabe: 4 (4 Seiten)

Punkte

#### 4.2.3 Zustandsdiagramm entwickeln

7



<b>Hauptprüfung 2012/2013</b>	<b>Berufliches Gymnasium (TG)</b>	
<b>1.5.2</b>	<b>Informationstechnik</b>	
<b>Lösungsvorschlag</b>	<b>Teil: 3 (Systeme)</b>	<b>Aufgabe: 5 (3 Seiten)</b>

5.1.1 2

100 Base TX: Ethernet Standard 100 Mbit/s, Twisted Pair, Basisband  
 Vollduplex: Senden und Empfangen eines Datenpaketes ist zur selben Zeit möglich.  
 Halbduplex: Gleichzeitiges Senden und Empfangen von Datenpaketen nicht möglich.

5.1.2 3

IP-Adresse 192.168.150.0/24

20 PCs → 5 Bits für Hostbereich, da  $2^5 = 32 > 20$

32 Bits - 24 Bits – 5 Bits = 3 Bits für Subnetze

SN-Mask /27 bzw. 255.255.255.224

SN1 (Verwaltung): SN-Adresse 192.168.150.0/27  
 Hosts 192.168.150.1/27 – 192.168.150.30/27

SN2 (Seminarraum): SN-Adresse 192.168.150.32/27  
 Hosts 192.168.150.33/27 – 192.168.150.62/27

SN3 (Bibliothek): SN-Adresse 192.168.150.64/27  
 Hosts 192.168.150.65/27 – 192.168.150.94/27

Bemerkung: Es sind auch 6 Bits für Hostbereich und 2 Bits für Subnetzbildung möglich.

5.1.3 1

Private Adressen wie 192.168.150.0 werden nicht ins Internet geroutet. Damit eine Kommunikation mit dem ISP hergestellt werden kann, muss dieser dem User eine gültige, öffentliche IP-Adresse zuweisen. NAT (Network Address Translation) dient dazu, die private IP-Adresse durch eine öffentliche IP-Adresse zu ersetzen.

5.1.4 2

Windows -> NTFS

- Verwendet einen MFT anstatt eines FAT, welche kleine Dateien direkt aufnehmen kann.
- MFT-Mirror in der Partitionsmitte => Sicherer als bei FAT.

Linux -> ext3

- Eine Datei kann unter verschiedenen Namen angesprochen werden (inodes)
- Journaling -> hohe Datensicherheit.

Hinweis: Jeweils nur ein Vorteil gefordert, weitere Vorteile möglich.

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik	
Lösungsvorschlag	Teil: 3 (Systeme)	Aufgabe: 5 (3 Seiten)

5.1.5

4

a) Anzahl der möglichen Adressen =  $2^{32}$  bzw.  $2^{28}$  (siehe Bemerkung unten)

-> Es lassen sich  $2^{32}$  Cluster adressieren.

-> bei einer Dateigröße kleiner als 4KiB sind das  $2^{32}$  Dateien.

$$\text{Theoretisch max. Kapazität} = 2^{32} * 32\text{KiB} = 2^{32} * 2^{15}\text{B} = 2^{47}\text{B} = 2^{37}\text{KiB} = 2^{27}\text{MiB} = 2^{17}\text{GiB} \\ = 2^7\text{TiB} = 128 \text{ TiB}$$

Bemerkung:

Es stehen nur 28 Bit zur Adressierung zur Verfügung (4Bit wurden „reserviert“). Dann ist die theoretische max. Kapazität 8 TiB.

Hinweis:

Name (Symbol)	Bedeutung	IEC-Name (IEC-Symbol)	Bedeutung
Kilobyte (kB)	$10^3$ Byte = 1000 Byte	Kibibyte (KiB)	$2^{10}$ Byte = 1024 Byte
Megabyte (MB)	$10^6$ Byte = 1000000 Byte	Mebibyte (MiB)	$2^{20}$ Byte = 1048576 Byte
Gigabyte (GB)	$10^9$ Byte = 1000000000 Byte	Gibibyte (GiB)	$2^{30}$ Byte = 1073741824 Byte

b.)

Datei	Startcluster	Dateigröße	Belegung	Cluster
Songs.txt	03	14 KiB	=> 16 KiB	4
Lyrics.txt	07	5 KiB	=> 8 KiB	2
Rechner.java	02	13 KiB	=> 16 KiB	4

02	03	04	05	06	07	08	09	10	11
09	04	05	06	eof	08	eof	10	11	eof

5.1.6

3

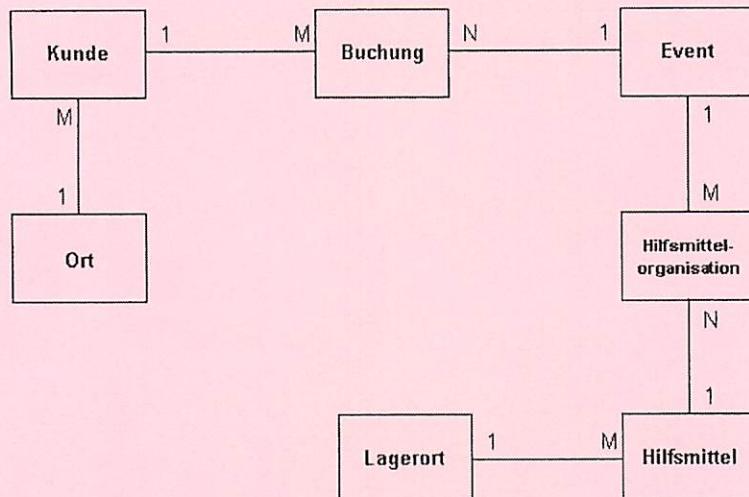
Die virtuelle Adresse muss 2 Bit länger als die physikalische Adresse sein, da 4mal so viel virtueller wie physikalischer Speicher zur Verfügung steht. Bei Seitengrößen von 4KiB ist die physikalische Größe auf 12 Bit festgelegt.

Eine virtuelle Adresse muss 12Bit+2Bit = 14Bit lang sein.

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik	
Lösungsvorschlag	Teil: 3 (Systeme)	Aufgabe: 5 (3 Seiten)

5.2.1

a.)



9

Bemerkung: Ort der Kunden kann auch als Attribut realisiert werden.

b.) Event

(ENr, Bezeichnung)

Kunde (KNr, ONr, Name, Vorname, Strasse, Telefon)

Hilfsmittel (HNr, LONr, Bezeichnung)

Lagerort (LONr, Bezeichnung)

Ort (ONr, PLZ, Ortsname)

Buchung (BNr, KNr, ENr, Datum, Uhrzeit)

Hilfsmittelorganisation (HNr, ENr, HNr) auch: ENr, HNr zusammen als PK

5.2.2.1 SELECT Speise.Bezeichnung, Preisgruppe.Preis

2

FROM Speise, Preisgruppe

WHERE Speise.PNr = Preisgruppe.PNr

ORDER BY Speise.Bezeichnung DESC;

5.2.2.2 SELECT Lieferant.name, SUM(Speisenlieferung.Anzahl\_Mahlzeiten)

4

FROM Lieferant, Speisenlieferung

WHERE Lieferant.LNr = Speisenlieferung.LNr

GROUP BY Lieferant.name

HAVING SUM(Speisenlieferung.Anzahl\_Mahlzeiten) > 100;

15

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)	
1.5.2	Informationstechnik (Systeme)	
Lösungsvorschlag	Teil: 3 (Wahlbereich)	Aufgabe: 6 (3 Seiten)

Punkte

6.1.1

2

Schulnetz	Verwaltungsnetz
Switch, OSI-Schicht 2	Hub (Repeater), OSI-Schicht 1
Voll duplex	Halbduplex
sicherer	unsicherer
Schneller	Langsamer wegen möglichen Kollisionen

Hinweis: Nur 2 Vorteile, nur für ein Gerät und Zuordnung beider zur OSI-Schicht verlangt.  
Weitere Vorteile möglich.

6.1.2

1

Bei einem Hub (Repeater), wie im Verwaltungsnetz, werden die Rahmen an alle anderen Stationen gesendet. Deswegen können alle „mithören“ und der Login-Name und das Passwort kann leicht, z.B. mit dem Programm „Wireshark“, mitgelesen werden („sniffen“).

6.1.3

4

a)

Class-C-Netz: 192.168.3.0 (andere Adressen aus dem Class-C Bereich möglich)

Netzmaske: /30 oder 255.255.255.252

=> 32 Bit - 30 Bit = 2 Bit für Hosts;  $2^2 - 2 = 2$  Adressen

Adressen an den Routern: 192.168.3.1 und 192.168.3.2 BC: 192.168.3.3

b) Routingtabelle:

	Netzadresse	Netzmaske	Gateway
1	192.168.1.0	255.255.255.0 (/24)	direkt
2	192.168.2.0	255.255.255.0 (/24)	direkt
3	192.168.3.0	255.255.255.252 (/30)	direkt
4	0.0.0.0	0.0.0.0 (/0)	192.168.3.2

Hinweis: Zeile 3 (Netzadresse) und Zeile 4 (Gateway) je nach Lösung aus Teil a).

6.1.4

3

Verzeichnisse	Unterverzeichnisse	Gruppe bzw. Benutzer	Rechte (lesen, schreiben)
Lehrer	Lehrer1	Lehrer1	lesen, schreiben
	Lehrer2	Lehrer2	lesen, schreiben
Schueler	Schueler1	Schueler1	lesen, schreiben
		Lehrer	lesen
	Schueler2	Schueler2	lesen, schreiben
		Lehrer	lesen
Tausch		Lehrer, Schueler	lesen, schreiben

<b>Hauptprüfung 2012/2013</b>	<b>Berufliches Gymnasium (TG)</b>	
<b>1.5.2</b>	<b>Informationstechnik (Systeme)</b>	
<b>Lösungsvorschlag</b>	<b>Teil: 3 (Wahlbereich)</b>	<b>Aufgabe: 6 (3 Seiten)</b>

**Punkte****6.1.5**

1

FAT 16 =>  $2^{16} = 65536$  Cluster möglich, Sektorgröße: 512 B  
=> max. Partitionsgröße =  $2^{16}$  Cluster \* 64 Sektoren/Cluster \* 512 B/Sektor  
= 2.147.483.648 B = 2.097.152 KiB = 2048 MiB = 2 GiB

Hinweis: 1 KiB =  $2^{10}$  B = 1024 B      1 kB =  $10^3$  B = 1000 B

**6.1.6**

2

<b>FAT</b>	<b>NTFS</b>
eingeschränkte Sicherheitsattribute.	Zugriffsschutz auf Verzeichnisse und Dateien.
Nach einem Absturz muss die gesamte Partition geprüft werden.	Nur die geöffneten Dateien müssen nach einem Absturz geprüft werden, da Journaling verwendet wird.
FAT benutzt eine simple Tabelle zum speichern der Dateieinträge .	NTFS verwendet eine Baumstruktur um seine Dateieinträge zu speichern.
Keine UNICODE Unterstützung.	Unterstützt UNICODE Zeichen im Dateinamen.

Hinweis: Nur 3 Unterschiede verlangt. Weitere Unterschiede möglich.

**6.1.7**

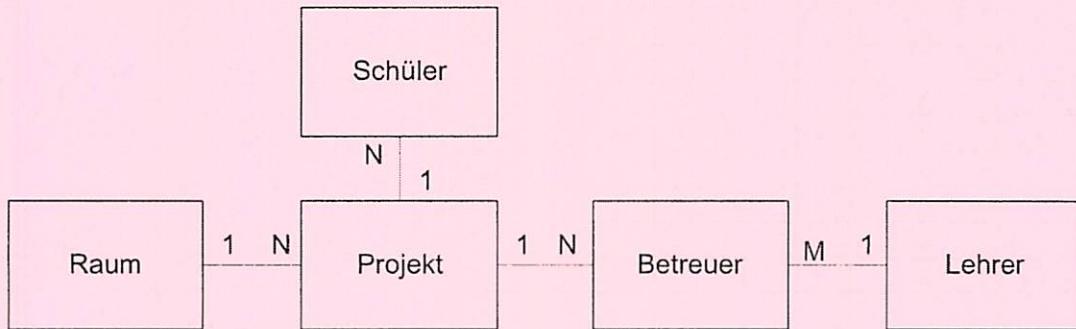
2

- Vollbackup: Alle Daten werden auf dem Backup-Medium gesichert.
- Differenzielles Backup: Gespeichert werden die Daten, die seit der letzten Vollsicherung erstellt oder verändert wurden. Für die Wiederherstellung werden die letzte Vollsicherung und das letzte differentielle Backup benötigt.
- Inkrementelles Backup: Gespeichert werden die Daten, die seit der letzten Sicherung erstellt oder verändert wurden. Für die Wiederherstellung werden die letzte Vollsicherung und alle inkrementellen Backups benötigt.

Hauptprüfung 2012/2013	Berufliches Gymnasium (TG)
1.5.2	Informationstechnik (Systeme)
Lösungsvorschlag	Teil: 3 (Wahlbereich)      Aufgabe: 6 (3 Seiten)

Punkte

6.2.1



Schüler(SID, PID, Name, Vorname, GebDatum)

8

Raum(RID, Bezeichnung)

Projekt(PID, RID, Bezeichnung, minAnzahl, maxAnzahl, Kostenbeitrag)

Betreuer(BID, PID, LID)      evtl. PID und LID zusammen als Primärschlüssel

Lehrer(LID, Name, Vorname)

Hinweis: **Primärschlüssel fett**, Fremdschlüssel unterstrichen

6.2.2.1

3

SELECT Name, Vorname, Telefon

FROM Ansprechpartner

WHERE Bereich LIKE "Obst"

OR Bereich LIKE "Getränke";

6.2.2.2

4

SELECT Lieferanten AS "Lieferanten Name", COUNT(\*) AS Anzahl

FROM Lieferanten l, Ansprechpartner a

WHERE l.Lief\_ID = a.Lief\_ID

GROUP BY Lieferanten;

<b>Hauptprüfung 2012/2013</b>	<b>Berufliches Gymnasium (TG)</b>			
<b>1.5.2</b>	<b>Informationstechnik</b>			
<b>Lösungsvorschlag</b>	<b>Teil: -</b>		<b>Aufgabe: - (1 Seite)</b>	

**Punkte**

**Übersicht: Punkteverteilung nach Anforderungsbereiche (AF: Anforderungsbereich)**

<b>Hardware Aufgabe 1</b>	<b>Punkte (Gesamt)</b>	<b>AF I</b>	<b>AF II</b>	<b>AF III</b>	<b>Hardware Aufgabe 2</b>	<b>Punkte (Gesamt)</b>	<b>AF I</b>	<b>AF II</b>	<b>AF III</b>
1.1.1	2		2		2.1.1	1	1		
1.1.2	1	1			2.1.2	1	1		
1.2.1	5		5		2.1.3	3	1	2	
1.2.2	1	1			2.1.4	5		1	4
1.2.3	2		2		2.1.5	2	2		
1.2.4	2		2		2.2.1	3		2	1
1.3.1	4	4			2.2.2	2		2	
1.3.2	1		1		2.2.3	2		2	
1.3.3	1		1		2.2.4	2		2	
1.4	4			4	2.3.1	3	2	1	
1.5.1	3	3			2.4.1	4		2	2
1.5.2	3		3		2.4.2	2		2	
1.5.3	1			1	Soll: 30:40:30	30	7	16	7
Soll: 30:40:30	30	9	16	5	Prozent:	100%	23%	53%	23%
Prozent:	100%	30%	53%	17%					

<b>Software Aufgabe 3</b>	<b>Punkte (Gesamt)</b>	<b>AF I</b>	<b>AF II</b>	<b>AF III</b>
3.1.1	2	1,5	0,5	
3.1.2	1	1		
3.2.1	2	1	1	
3.2.2	3	2	1	
3.2.3	3	1,5	1,5	
3.3	2	1	1	
3.4	5	1	3	1
3.5.1	3		2	1
3.5.2	2			2
3.6	7		2	5
Soll: 30:40:30	30	9	12	9
Prozent:	100%	30%	40%	30%

<b>Systeme Aufgabe 5</b>	<b>Punkte (Gesamt)</b>	<b>AF I</b>	<b>AF II</b>	<b>AF III</b>
5.1.1	2	2		
5.1.2	3	3		
5.1.3	1	1		
5.1.4	2	2		
5.1.5	4		4	
5.1.6	3		3	
5.2.1	9		4	5
5.2.2.1	2		2	
5.2.2.2	4			4
Soll: 30:40:30	30	8	13	9
Prozent:	100%	27%	43%	30%

<b>Software Aufgabe 4</b>	<b>Punkte (Gesamt)</b>	<b>AF I</b>	<b>AF II</b>	<b>AF III</b>
4.1.1	1	1		
4.1.2	2	1	1	
4.1.3	2			2
4.1.4	1			1
4.1.5	2			2
4.2.1.1	4		4	
4.2.1.2	3		3	
4.2.2	8			3
4.2.3	7			3
Soll: 30:40:30	30		9	12
Prozent:	100%		30%	40%

<b>Systeme Aufgabe 6</b>	<b>Punkte (Gesamt)</b>	<b>AF I</b>	<b>AF II</b>	<b>AF III</b>
6.1.1	2	2		
6.1.2	1			1
6.1.3	4	1	1	2
6.1.4	3	1	1	1
6.1.5	1			1
6.1.6	2		2	
6.1.7	2	1	1	
6.2.1	8	1	3	4
6.2.2.1	3	1	2	
6.2.2.2	4	1	3	
Soll: 30:40:30	30		10	12
Prozent:	100%		33%	40%