



Informatik

Leistungskurs

Beispielset 2

Auswahlverfahren: **siehe Seite 2**

Einlese- und Auswahlzeit: **30 Minuten**

Bearbeitungszeit: **240 Minuten**

Erlaubte Hilfsmittel:	Wörterbuch zur deutschen Rechtschreibung Hessisches Datenschutzgesetz
------------------------------	--

Sonstige Hinweise:	keine
---------------------------	--------------

Auswahlverfahren

Die Prüfungsaufgaben sind in die drei Gruppen A, B und C eingeteilt.

- Aus den beiden Aufgaben der Gruppe A wählt die zuständige Lehrkraft gemäß dem im Unterricht benutzten Sprachkonzept eine Aufgabe aus.
- Die Prüfungsteilnehmerinnen und Prüfungsteilnehmer entscheiden sich für eine der beiden Wahlaufgaben aus Gruppe B.
- Die Aufgabe aus Gruppe C ist eine Pflichtaufgabe.

Insgesamt sind drei Aufgaben zu bearbeiten, je eine aus den Gruppen A, B und C.

I. Thema und Aufgabenstellung

- Gruppe A** Auswahl einer Aufgabe durch die zuständige Lehrkraft
Gruppe B Auswahl einer Aufgabe durch die Prüfungsteilnehmerinnen und Prüfungsteilnehmer
Gruppe C Pflichtaufgabe

Aufgaben

Gruppe A

A 1 Index für Datenbanktabelle (Delphi/Pascal, Σ 35 BE)

Die Ausführung von SQL-Select-Anweisungen kann durch den Einsatz eines Indexes für eine Datenbanktabelle beschleunigt werden. Man kann für jedes beliebige Attribut einen Index erzeugen und verwalten, besonders sinnvoll ist dies aber für den jeweiligen Primärschlüssel. Im folgenden Beispiel soll für den Primärschlüssel FilmID ein Index gebildet werden:

Datenbanktabelle

G28	Findet Nemo	Andrew Stanton	Zeichentrick	...
X17	Herr der Ringe	Peter Jackson	Action	
B563	Last Samurai	Edward Zwick	Epos	
U47	Sams in Gefahr	Ben Verbong	Kinderfilm	
A18	Master&Commander	Peter Weir	Abenteuer	
E11	Freaky Friday	Mark Waters	Komödie	
K7	S.W.A.T.	Clark Johnson	Action	

Index als Feld

A18	5
B563	3
E11	6
G28	1
K7	7
U47	4
X17	2

- a) Ein Index lässt sich als sortiertes Feld mit Filmnummern und jeweiliger Position in der Datenbanktabelle oder als binärer Suchbaum aufbauen. Zeichnen Sie den Index als binären Suchbaum und vergleiche die beiden Ansätze. (6 BE)
- b) Analysieren Sie den Quelltext in der Anlage zur Aufgabe. Beschreiben Sie dann die Klassen und stellen Sie die Klassen, Attribute und Methoden in einem UML-Klassendiagramm dar. (14 BE)
- c) Analysieren und erläutern Sie die beiden folgenden Methoden der Klasse TBinBaum: (6 BE)

```

procedure TBinBaum.einfuegenBaum(amKnoten, neuerKnoten: TKnoten);
begin
    if neuerKnoten.filmID < amKnoten.filmID then

```

```
    if amKnoten.toLinks = nil
    then amKnoten.toLinks:= neuerKnoten
    else einfuegenBaum(amKnoten.toLinks, neuerKnoten)
else
    if amKnoten.toRechts = nil
    then amKnoten.toRechts:= neuerKnoten
    else einfuegenBaum(amKnoten.toRechts, neuerKnoten);
end;

procedure TBinBaum.einfuegen(filmID: string; position: integer);
    var Knoten: TKnoten;
begin
    Knoten:= neuerKnoten(filmID, position);
    if Wurzel = nil
    then Wurzel:= Knoten
    else einfuegenBaum(Wurzel, Knoten)
end;
```

d) Das Ergebnis der SQL-Anweisung

```
SELECT Kategorie FROM Film WHERE FilmID = 'U47'
```

soll unter Nutzung des Indexes ermittelt werden. Entwerfen und implementieren Sie geeignete Methoden, um eine solche SELECT-Anweisung berechnen zu können. (9 BE)

Anlage

```
TDatensatz = class
    filmID: string;
    titel: string;
    regisseur: string;
    kategorie: string;
    constructor create(f, t, r, k: string);
end;
```

```
constructor TDatensatz.create(f, t, r, k: string);
begin
    filmID := f;
    titel := t;
    regisseur:= r;
    kategorie:= k;
end;
```

```
TDBTabelle = class
    index: TIndex;
    anzahlDatensaetze: integer;
    datensaetze: array of TDatensatz;

    constructor create(groesse: integer);
    procedure einfuegen(einDatensatz: TDatensatz);
    function gibDatensatz(i: integer): TDatensatz;
    procedure erzeugeIndex;
end;
```

```
TIndex = class(TBinBaum)
    procedure erzeugeFuer(Tabelle: TDBTabelle);
```

```
function gibPosition(filmID: string): integer;
end;

procedure TIndex.erzeugeFuer(tabelle: TDBTabelle);
  var einDatensatz: TDatensatz;
      i: integer;
begin
  for i:= 0 to tabelle.anzahlDatensaetze-1 do begin
    einDatensatz:= tabelle.gibDatensatz(i);
    einfuegen(einDatensatz.filmID, i);
  end;
end;

TKnoten = class
  filmID : string;
  position: Integer;
  toLinks : TKnoten;
  toRechts: TKnoten;
  constructor create(filmID: string; position: integer);
end;

TBinBaum = class
  Wurzel: TKnoten;
  constructor create;
  procedure einfuegen(filmID: string; position: integer);
  ...
end;
```

A 2 Index für Datenbanktabelle (Java, Σ 35 BE)

Die Ausführung von SQL-Select-Anweisungen kann durch den Einsatz eines Indexes für eine Datenbanktabelle beschleunigt werden. Man kann für jedes beliebige Attribut einen Index erzeugen und verwalten, besonders sinnvoll ist dies aber für den jeweiligen Primärschlüssel. Im folgenden Beispiel soll für den Primärschlüssel FilmID ein Index gebildet werden:

Datenbanktabelle

G28	Findet Nemo	Andrew Stanton	Zeichentrick	...
X17	Herr der Ringe	Peter Jackson	Action	
B563	Last Samurai	Edward Zwick	Epos	
U47	Sams in Gefahr	Ben Verbong	Kinderfilm	
A18	Master&Commander	Peter Weir	Abenteuer	
E11	Freaky Friday	Mark Waters	Komödie	
K7	S.W.A.T.	Clark Johnson	Action	

Index als Feld

A18	5
B563	3
E11	6
G28	1
K7	7
U47	4
X17	2

- a) Ein Index lässt sich als sortiertes Feld mit Filmnummern und jeweiliger Position in der Datenbanktabelle oder als binärer Suchbaum aufbauen. Zeichnen Sie den Index als binären Suchbaum und vergleiche die beiden Ansätze. (6 BE)
- b) Analysieren Sie den Quelltext in der Anlage zur Aufgabe. Beschreiben Sie dann die Klassen und stellen Sie die Klassen, Attribute und Methoden in einem UML-Klassendiagramm dar. (14 BE)
- c) Analysieren und erläutern Sie die beiden folgenden Methoden der Klasse BinBaum: (6 BE)

```
private void einfuegenBaum(BinKnoten myBaum, BinKnoten neuKnoten) {
    if (neuKnoten.filmID.compareTo(myBaum.filmID) < 0)
        if (myBaum.links != null)
            einfuegenBaum(myBaum.links, neuKnoten);
        else
            myBaum.links = neuKnoten;
    else
        if (myBaum.rechts != null)
            einfuegenBaum(myBaum.rechts, neuKnoten);
        else
            myBaum.rechts = neuKnoten;
}

public void einfuegen(String filmID, int position) {
    BinKnoten neuKnoten = new BinKnoten(filmID, position);
    if (wurzel == null)
        wurzel = neuKnoten;
    else
        einfuegenBaum(wurzel, neuKnoten);
}
```

- d) Das Ergebnis der SQL-Anweisung

```
SELECT Kategorie FROM Film WHERE FilmID = 'U47'
```

soll unter Nutzung des Indexes ermittelt werden. Entwerfen und implementieren Sie geeignete Methoden, um eine solche SELECT-Anweisung berechnen zu können. (9 BE)

Anlage

```
class Datensatz {
    String filmID;
    String titel;
    String regisseur;
    String kategorie;

    public Datensatz(String f, String t, String r, String k) {
        filmID = f;
        titel = t;
        regisseur = r;
        kategorie = k;
    }
}

public class DBTabelle extends Vector {

    Index einIndex;
    Datensatz einDatensatz;

    public void einfuegen(Datensatz einDatensatz) {...}
    public Datensatz gibDatensatz(int i) {...}
    public void erzeugeIndex() {...}

}

public class Index extends BinBaum {

    public Index(DBTabelle eineTabelle) {
        Datensatz einDatensatz;

        for ( int i = 0 ; i < eineTabelle.size() ; i++ ) {
            einDatensatz = eineTabelle.gibDatensatz(i);
            einfuegen(einDatensatz.filmID, i);
        }
    }

}

class BinKnoten {
    ...
}

public class BinBaum {
    BinKnoten wurzel;

    public BinBaum() {...}
    public void einfuegen(String filmID, int position) {...}
    ...
}
```

Gruppe B

B 1 Datenbank Busunternehmen (Σ 35 BE)

Ein Busunternehmen bietet Tagesausflüge an. Stammkunden erhalten halbjährlich das Ausflugsprogramm, neue Kunden werden durch Annoncen in der Tageszeitung geworben. Für jede Fahrt wird höchstens ein Bus eingesetzt. Die Organisation der Ausflüge soll mit Hilfe einer Datenbank erfolgen, in der folgende Daten enthalten sein müssen: von jedem Bus, den das Unternehmen besitzt, das Kfz-Kennzeichen, der Bustyp, die Anzahl der Sitzplätze und das Baujahr, von jeder Fahrt das Reiseziel, das Datum, die Dauer in Stunden und der Preis pro Person, von jedem Fahrer des Unternehmens Name und Anschrift (Plz, Ort, StrasseNr, Telefon), dabei sei vorausgesetzt, dass es nicht zwei Fahrer mit demselben Namen in dem Busunternehmen gibt, von den Kunden Name und Anschrift (Plz, Ort, StrasseNr, Telefon) und ihre gebuchten Fahrten mit Angabe der für sie reservierten Plätze.

- a) Modellieren Sie eine Datenbank zur Verwaltung des Busunternehmens als ER-Diagramm und erläutern Sie die darin vorkommenden Beziehungen. (9 BE)
- b) Überführen Sie das ER-Diagramm in ein relationales Datenbankschema und kennzeichnen Sie die Primär- und Fremdschlüssel. (5 BE)
- c) Ein unerfahrener Busunternehmer hat unter anderem folgende Relation entworfen: (5 BE)

Fahrt (FahrtNr, Ziel, Datum, Dauer, Preis, Kfz, Sitzplätze, Bustyp)

- 1) Erläutern Sie anhand von zwei Beispielen welche Probleme dabei auftreten können.
 - 2) Begründen Sie, dass sich die Relation in der zweiten, aber nicht in der dritten Normalform befindet. Bringen Sie die Relation aus c in die 3. Normalform.
- d) Erläutern Sie die folgenden SQL-Befehle: (6 BE)
- 1) insert into bus values ('DA-HT-4354', 'Mercedes Benz MB1200S', 66, 1988);
 - 2) select Reiseziel, Datum, Sitzplaetze
from Bus, Fahrt
where (Bus.Kfz = Fahrt.Kfz);
 - 3) select Fahrer.Name, Sum(Dauer)
from Fahrer, Fahrten
where (Datum between #1/03/05# AND #31/03/05#) and
(Fahrer.Name = Fahrten.Name)
group by Fahrer.Name;
- e) Geben Sie SQL-Anweisungen für folgende Abfragen an: (10 BE)
- 1) Teilnehmerliste für die Fahrt nach Straßburg am 14.7.2005.
 - 2) Liste mit Anzahl der freien Plätze für Fahrten ab dem 31.1.2005.

B 2 Datenbank Kino (Σ 35 BE)

Ein Kino verfügt über mehrere Säle mit unterschiedlicher Anzahl von Sitzplätzen. Die Sitzplätze in einem Saal sind zu Verwaltungszwecken aufsteigend nummeriert. Für jeden Sitzplatz sind auch die Kategorie (Loge bzw. Parkett), die Reihe und seine spezielle Platznummer in der Reihe angegeben. In den Sälen finden die Vorstellungen statt, die durch eindeutige Vorstellungsnummern identifiziert werden. Zu jeder Vorstellung soll die Gesamtzahl der bisher verkauften Kinokarten gespeichert werden. Das Kinoprogramm gibt alle Vorstellungen nach Datum, Uhrzeit und gezeigtem Film an. Neben den Filmtiteln sind Regisseur, Dauer, Kategorie (Action, Komödie, ...) und FSK-Altersangabe wichtig. Auf jeder Kinokarte werden Reihe und Platznummer des gebuchten Sitzplatzes angegeben.

- a) Modellieren Sie eine Datenbank für das Kino als ER-Diagramm, analysieren und beschreiben Sie die im ER-Diagramm vorkommenden Beziehungen. **(10 BE)**
- b) Übertragen Sie das ER-Diagramm in ein optimiertes Relationenmodell und erläutern Sie das Primärschlüsselproblem der Sitzplatztabelle. **(4 BE)**
- c) Erläutere sowohl SQL-technisch als auch anwendungsbezogen die beiden folgenden SQL-Anweisungen: **(6 BE)**

c1)

```
SELECT Datum, Uhrzeit, Titel, FSK
FROM Vorstellung, Film
WHERE Vorstellung.FilmID = Film.FilmID AND Datum = #10/04/2004#
ORDER BY Uhrzeit
```

c2)

```
SELECT VorstellungID, Datum, Titel
FROM Vorstellung, Film
WHERE Vorstellung.FilmID = Film.FilmID AND
Verkauft = (
    SELECT MAX(Verkauft)
    FROM Vorstellung, Film
    WHERE Vorstellung.FilmID = Film.FilmID AND Kategorie = 'Action' )
```

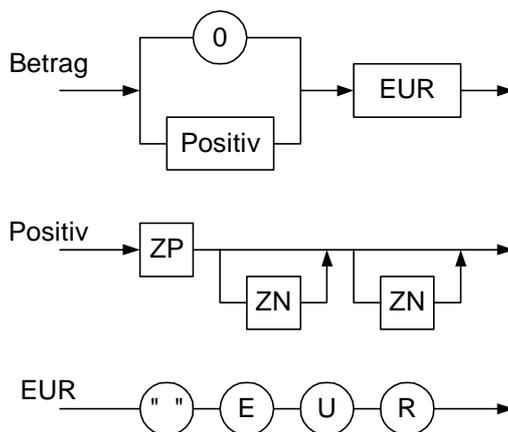
- d) Formuliere SQL-Anweisungen für folgende Fragestellungen: **(9 BE)**
 - d1) Gesucht ist eine Liste aller Abenteuer-Filme die im Monat Mai laufen.
 - d2) Gesucht ist eine Liste der nicht ausverkauften Vorstellungen von *Herr der Ringe III*.
 - d3) Gesucht ist eine Liste mit den Gesamtzahlen aller verkauften Kinokarten pro Film für die Filme mit mindestens 1000 Besuchern.
- e) Über das Internet können Einzelplatz-Reservierungen von Vorstellungen vorgenommen werden. Man gibt dabei seinen Namen und die Sitzplatz-Kategorie an und erhält als Bestätigung eine Reservierungsnummer, die man an der Kinokasse vorlegen muss. Beschreibe die benötigte Datenbank-Erweiterung und bewerte das Online-Reservierungssystem unter technischen und gesellschaftlichen Aspekten. **(6 BE)**

Gruppe C

C 1 Geldbeträge

Im Folgenden geht es um eine Grammatik für Geldbeträge in der Währung Euro. Dabei wird die im internationalen Zahlungsverkehr verwendete Abkürzung EUR benutzt.

- a) Erläutern Sie die drei Syntaxdiagramme für Geldbeträge, beschreiben Sie die durch sie definierte Sprache und erweitern Sie die Syntaxdiagramme so, dass optional zweistellige Kommabeträge möglich sind. **(6 BE)**



ZP steht für die Ziffern von 1 bis 9 und ZN für die Ziffern von 0 bis 9.

- b) Zeichnen Sie das Zustandsdiagramm eines endlichen Automaten, der Geldbeträge nach Aufgabe a mit optionalem zweistelligen Kommabetrag erkennt. **(6 BE)**
- c) Übersetzen Sie das Zustandsdiagramm in eine äquivalente Grammatik und geben Sie eine Ableitung für das Wort 546,80 EUR an. **(7 BE)**
- d) Erläutern Sie den folgenden Akzeptor: **(4 BE)**
- ```

akzeptiere_positiv(Ein, Aus):-
 akzeptiere_zp(Ein, L1),
 akzeptiere_zn(L1, L2),
 akzeptiere_zn(L2, Aus).

akzeptiere_zp([K|R], R):-
 member(K, ['1', '2', '3', '4', '5', '6', '7', '8', '9']).

akzeptiere_zn([K|R], R):-
 member(K, ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']).
 akzeptiere_zn(R, R).

```
- e) Erweitern Sie Aufgabe d zu einem Akzeptor für Betrag mit optionalem zweistelligen Kommabetrag. **(7 BE)**

## Korrektur- und Bewertungshinweise - nicht für den Prüfungsteilnehmer bestimmt -

### II. Erläuterungen, Lösungshinweise, Bewertung und Beurteilung

#### A 1 Index für Datenbanktabelle – Delphi/Pascal

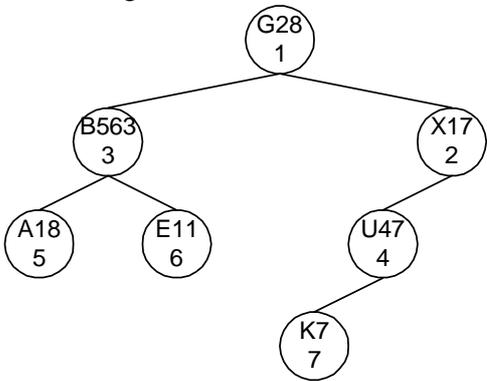
##### Aufgabenart

Die Aufgabe gehört zum Halbjahresthema 12.1 Objektorientierte Modellierung. Sie erfordert den Vergleich zwischen sortiertem Feld und binärem Suchbaum, die Analyse und Darstellung einer objektorientierten Modellierung, die Implementierung von Klassen und Methoden, sowie die Analyse und Synthese von Methoden für binäre Suchbäume im kursübergreifenden, anwendungsorientierten Zusammenhang.

##### Voraussetzungen des Lehrplans

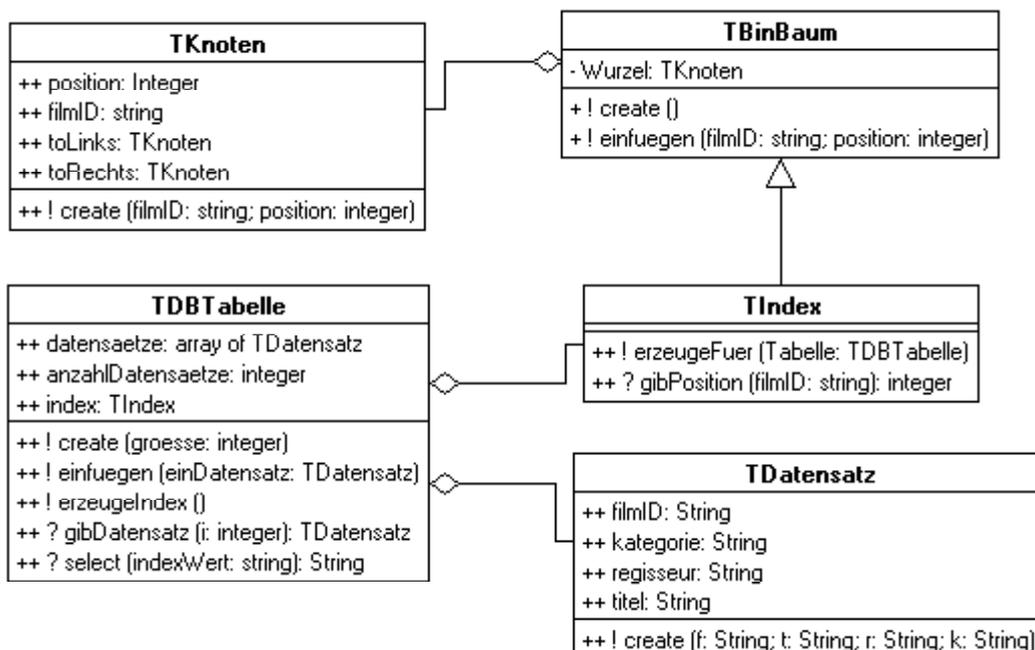
Von den verbindlichen Inhalten des Lehrplans werden zur Lösung dieser Aufgabe folgende Inhalte benötigt: Darstellung von Klassen, Objektbeziehungen und Vererbung mit der grafischen Modellierungssprache UML, Vererbung und Klassenhierarchie, rekursive und iterative Verfahren, binärer Suchbaum sowie SQL-Abfragen.

##### Bewertungseinheiten

| erwartete Schülerleistung |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | I | II | III |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|----|-----|
| a                         | <p>Darstellung des Index als binärer Suchbaum</p>  <pre> graph TD     G28((G28<br/>1)) --- B563((B563<br/>3))     G28 --- X17((X17<br/>2))     B563 --- A18((A18<br/>5))     B563 --- E11((E11<br/>6))     X17 --- U47((U47<br/>4))     U47 --- K7((K7<br/>7))                     </pre> <p>Vergleich der beiden Ansätze hinsichtlich Komplexität der Operationen Einfügen, Suchen und Löschen (<math>O(n)</math> bzw. <math>O(\log n)</math>)</p> | 3 |    |     |
| b                         | Analyse und Beschreibung der Klassen, dazu sind gute Kenntnis über objektorientierte Modellierung, insbesondere im Zusammenhang mit Vererbung erforderlich. Darstellung der Klassen, Attribute und Methoden als UML-Klassendiagramm (siehe unten)                                                                                                                                                                                                                                                                                      | 5 | 9  |     |
| c                         | Analyse und Beschreibung von zwei Methoden der TBinBaum-Klasse. Es sind sowohl die implementierungsspezifischen Details als auch der Zusammenhang mit der Problemstellung der Aufgabe zu erläutern.                                                                                                                                                                                                                                                                                                                                    |   | 6  |     |
| d                         | Entwurf und Implementierung von Methoden zur Berechnung einer SELECT-Anweisung. Gebraucht wird eine Methode der Klasse TBinBaum zum Suchen im binären Suchbaum                                                                                                                                                                                                                                                                                                                                                                         |   | 3  | 6   |

| erwartete Schülerleistung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | I  | II | III |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|----|-----|
| <pre> <b>function</b> TBinBaum.suche(filmID: <b>string</b>): integer;   <b>var</b> Knoten: TKnoten; <b>begin</b>   Knoten:= Wurzel;   <b>while</b> Knoten &lt;&gt; nil <b>do begin</b>     <b>if</b> filmID &lt; Knoten.filmID       <b>then</b> Knoten:= Knoten.toLinks     <b>else if</b> filmID &gt; Knoten.filmID       <b>then</b> Knoten:= Knoten.toRechts     <b>else begin</b>       result:= Knoten.position;       exit;     <b>end;</b>   <b>end;</b>   result:= -1; <b>end;</b> </pre> <p>und eine Methode zur Auswertung der SELECT-Anweisung:</p> <pre> <b>function</b> TDBTabelle.select(indexWert: <b>string</b>): <b>string</b>;   <b>var</b> pos: integer;       einDatensatz: TDatensatz; <b>begin</b>   pos:= index.suche(indexWert);   <b>if</b> pos &gt;= 0 <b>then begin</b>     einDatensatz:= gibDatensatz(pos);     result:= einDatensatz.kategorie;   <b>end</b>   <b>else</b>     Result:= ''; <b>end;</b> </pre> |    |    |     |
| <b>Σ 40 BE</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | 11 | 18 | 6   |

**Mögliches UML-Diagramm zu Aufgabe b:**



## A 2 Index für Datenbanktabelle – Java

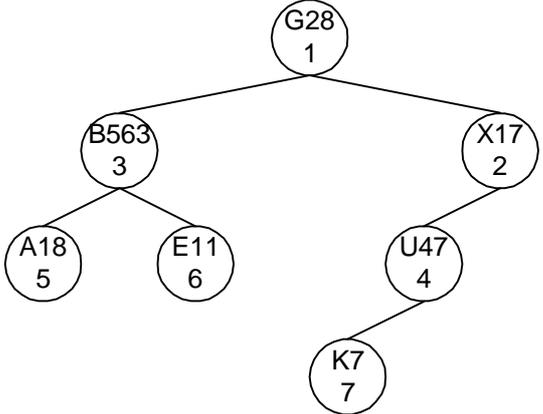
### Aufgabenart

Die Aufgabe gehört zum Halbjahresthema 12.1 Objektorientierte Modellierung. Sie erfordert den Vergleich zwischen sortiertem Feld und binärem Suchbaum, die Analyse und Darstellung einer objektorientierten Modellierung, die Implementierung von Klassen und Methoden, sowie die Analyse und Synthese von Methoden für binäre Suchbäume im kursübergreifenden, anwendungsorientierten Zusammenhang.

### Voraussetzungen des Lehrplans

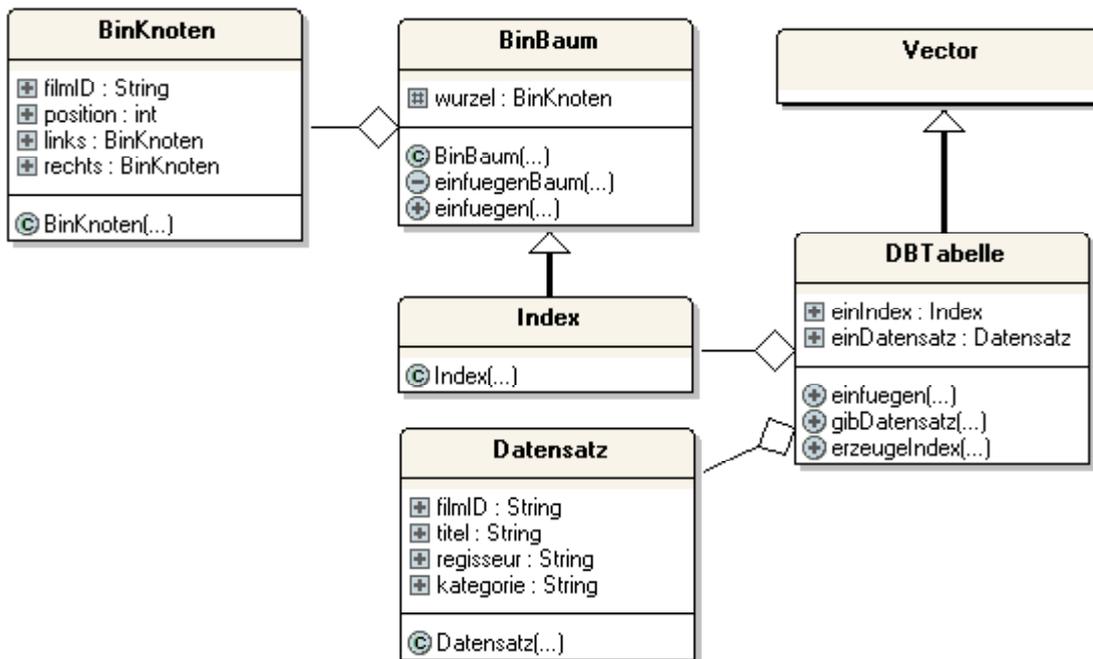
Von den verbindlichen Inhalten des Lehrplans werden zur Lösung dieser Aufgabe folgende Inhalte benötigt: Darstellung von Klassen, Objektbeziehungen und Vererbung mit der grafischen Modellierungssprache UML, Vererbung und Klassenhierarchie, rekursive und iterative Verfahren, binärer Suchbaum sowie SQL-Abfragen.

### Bewertungseinheiten

|   | erwartete Schülerleistung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | I | II | III |
|---|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|----|-----|
| a | <p>Darstellung des Index als binärer Suchbaum</p>  <pre> graph TD     G28((G28<br/>1)) --- B563((B563<br/>3))     G28 --- X17((X17<br/>2))     B563 --- A18((A18<br/>5))     B563 --- E11((E11<br/>6))     X17 --- U47((U47<br/>4))     U47 --- K7((K7<br/>7))                     </pre> <p>Vergleich der beiden Ansätze hinsichtlich Komplexität der Operationen Einfügen, Suchen und Löschen: <math>O(n)</math> bzw. <math>O(\log n)</math></p> | 3 |    |     |
| b | Analyse und Beschreibung der Klassen, dazu sind gute Kenntnis über objektorientierte Modellierung, insbesondere im Zusammenhang mit Vererbung erforderlich. Darstellung der Klassen, Attribute und Methoden als UML-Klassendiagramm (siehe unten)                                                                                                                                                                                                                                                                                    | 5 | 9  |     |
| c | Analyse und Beschreibung von zwei Methoden der BinBaum-Klasse. Es sind sowohl die implementierungsspezifischen Details als auch der Zusammenhang mit der Problemstellung der Aufgabe zu erläutern.                                                                                                                                                                                                                                                                                                                                   |   | 6  |     |
| d | <p>Entwurf und Implementierung von Methoden zur Berechnung einer SELECT-Anweisung.<br/>Gebraucht wird eine Methode der Klasse BinBaum zum Suchen im binären Suchbaum</p> <pre> public int suche(String filmID) {     BinKnoten einKnoten = wurzel;     while (einKnoten != null) {         int vergleich = filmID.compareTo(einKnoten.filmID);         if (vergleich &lt; 0)                     </pre>                                                                                                                              |   | 3  | 6   |

| erwartete Schülerleistung                                                                                                                                                                                                                                                                                                                                                                                                                         | I  | II | III |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|----|-----|
| <pre> einKnoten = einKnoten.links; else if (vergleich &gt; 0)     einKnoten = einKnoten.rechts; else     return einKnoten.position; } return -1; }  und eine Methode zur Auswertung der SELECT-Anweisung:  public String select(String indexWert) {     int pos = einIndex.suche(indexWert);     if (pos &gt;= 0) {         einDatensatz = gibDatensatz(pos);         return einDatensatz.kategorie;     }     else         return null; } </pre> |    |    |     |
| Σ 35 BE                                                                                                                                                                                                                                                                                                                                                                                                                                           | 11 | 18 | 6   |

Mögliches UML-Diagramm zu Aufgabe b:



## B 1 Datenbank Busunternehmen

### Aufgabenart

Die Aufgabe gehört zum Halbjahresthema 12.2 Datenbanken. Die Prüflinge müssen die beschriebene Realität selbstständig modellieren und das Modell in einem vollständigen ER-Diagramm darstellen. Dabei sind die Beziehungen und deren Kardinalität und Optionalität zu erläutern. Das sich daraus ergebende Datenbankschema muss hinsichtlich der Anzahl der Tabellen optimiert werden. Für die Schüler muss die Normalisierung bis zur dritten Normalform verfügbar sein und zur Vermeidung von Einfüge- und Löschanomalien angewendet werden können. Die gegebenen SQL Befehle müssen in Bezug auf den Verwendungszweck erläutert werden. Aufgabe e2 erfordert ein tieferes Verständnis der Abfragemöglichkeiten mit SQL.

### Voraussetzungen des Lehrplans

Zur Lösung der Aufgabe werden folgende verbindlichen Inhalte des Lehrplans benötigt: ER-Modell mit Objekten, Beziehungen und Kardinalitäten, Abbildung des ER-Modells in das Relationenmodell, Realisierung in einem Datenbanksystem, Normalisierung, SQL mit DQL und DML mit Aggregatfunktionen.

### Bewertungseinheiten

|   | erwartete Schülerleistung                                                                                                                                                                                                                                                                                                                                                              | I | II | III |
|---|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|----|-----|
| a | <p>obligatorische 1:n –Beziehung zwischen Bus und Fahrt<br/>obligatorische 1:n –Beziehung zwischen Fahrer und Fahrt<br/>optionale m:n – Beziehung zwischen Fahrt und Kunde</p>                                                                                                                                                                                                         | 3 | 6  |     |
| b | <p>Optimiertes Datenbankschema, Kennzeichnung von Primär- und Fremdschlüssel:</p> <p>Bus (<u>Kfz</u>, Bustyp, Sitzplätze, Baujahr)</p> <p>Fahrer (<u>Name</u>, StrasseNr, PLZ, Ort, Telefon)</p> <p>Fahrt (<u>FNr</u>, Reiseziel, Datum, Dauer, Preis, ↑KFZ, ↑Name)</p> <p>Kunde (<u>KNr</u>, Name, StrasseNr, PLZ, Ort, Telefon)</p> <p>Buchung(↑<u>FNr</u>, ↑<u>KNr</u>, Plätze)</p> |   | 5  |     |
| c | <p>Integritätsprobleme durch Einfüge- und Löschanomalien.</p> <p>Die Relation ist in der 1. NF, weil alle Attribute atomar sind.</p> <p>Die Relation ist in der 2. NF, weil die Relation in der 1. NF ist und alle Nicht-Schlüsselattribute nur vom Primärschlüssel abhängen.</p>                                                                                                      | 2 |    |     |

| erwartete Schülerleistung |                                                                                                                                                                                                                                                                                                                                                                                                                                        | I           | II | III |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|----|-----|
|                           | Die Relation ist nicht in der 3. NF. Es besteht eine transitive Abhängigkeit zwischen Bustyp, Kennzeichen und Primärschlüssel.                                                                                                                                                                                                                                                                                                         |             | 3  |     |
| d                         | Einfügen eines neuen Datensatzes mit DA-HT-4354, Mercedes Benz usw.<br>Für jedes Reiseziel, Datum wird die Anzahl der Sitzplätze ausgegeben.<br>Ausgabe einer Liste mit den summierten Fahrtzeiten der Fahrer im März 2005                                                                                                                                                                                                             | 2<br>2<br>2 |    |     |
| e                         | 1<br>SELECT Name, Plätze<br>FROM Kunde, Buchung, Fahrt<br>WHERE (Kunde.KNr = Buchung.KNr) AND<br>(Buchung.FNr = Fahrt.FNr) AND<br>(Reiseziel = 'Strassburg') AND<br>(Datum = #14/07/05#);<br>2<br>SELECT Reiseziel, Datum,<br>Sitzplätze-Sum(Platze) AS FreiePlätze<br>FROM Bus, Fahrt, Buchung<br>WHERE (Bus.Kfz = Fahrt.Kfz) AND<br>(Fahrt.FNr = Buchung.FNr) AND<br>(Datum >= #31/01/05#)<br>GROUP BY Reiseziel, Datum, Sitzplätze; |             | 4  | 6   |
|                           | <b>Σ 35 BE</b>                                                                                                                                                                                                                                                                                                                                                                                                                         | 11          | 18 | 6   |

## B 2 Datenbank Kino

### Aufgabenart

In der Aufgabe sind die Modellierung einer Datenbank als ER-Diagramm, die Umsetzung in das Relationenmodell, die Analyse und Konstruktion von SQL-Anweisungen sowie die Analyse von Web-Datenbanken gefordert.

### Voraussetzungen des Lehrplans

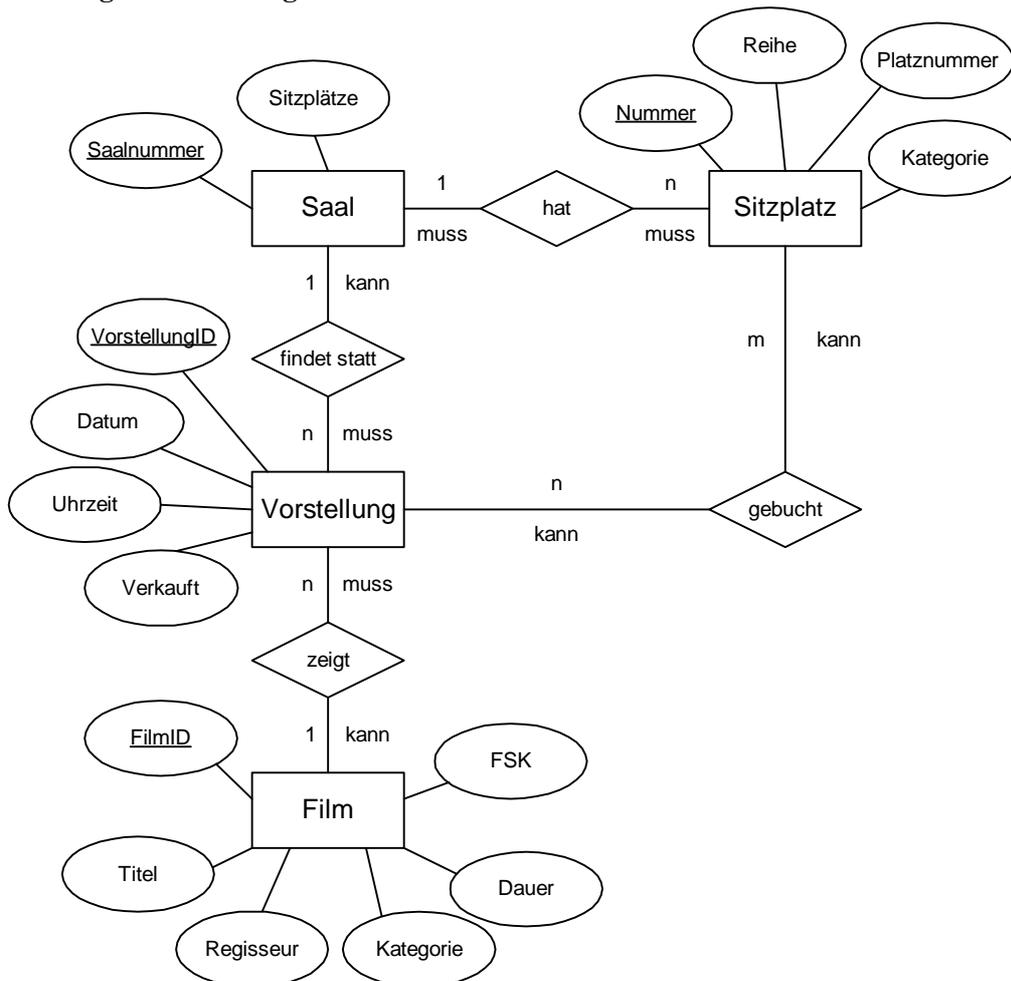
Von den verbindlichen Inhalten des Lehrplans werden zur Lösung der Aufgabe benötigt: die Modellierung von Mini-Welten mit ER-Diagrammen, die Umsetzung eines ER-Diagramms in das Relationenmodell, die Abfragesprache SQL, Datensicherheit und Datenschutz.

### Bewertungseinheiten

|    | erwartete Schülerleistung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | I | II | III |
|----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|----|-----|
| a  | <p>Das zu entwerfende ER-Diagramm muss die vier Objekttypen <i>Saal</i>, <i>Sitzplatz</i>, <i>Vorstellung</i> und <i>Film</i> enthalten, sowie die vier Beziehungen <i>findet statt</i>, <i>hat</i>, <i>gebucht</i> und <i>zeigt</i>. Primärschlüssel sind zu unterstreichen, bei der Gestaltung der Attribute gibt es Freiheiten. Die im Text genannten Attribute müssen vorhanden sein.</p> <p>ER-Diagramm siehe unten</p> <p>Kardinalität und Optionalität sind im ER-Diagramm dargestellt. Diese sind sprachlich zu erläutern, z. B. ein Film kann in mehreren Vorstellungen gezeigt werden, daher Kardinalität n, muss aber nicht gezeigt werden, daher Kann-Beziehung. Eine Vorstellung muss genau einen Film zeigen, daher Kardinalität 1 und Muss-Beziehung.</p> | 3 | 3  |     |
| b  | <p>Saal(<u>Saalnummer</u>, Sitzplätze)</p> <p>Sitzplatz(<u>Nummer</u>, Reihe, Platznummer, Kategorie, ↑<u>Saalnummer</u>)</p> <p>Film(<u>FilmNr</u>, Titel, Regisseur, Katgeorie, Dauer, FSK)</p> <p>Vorstellung(<u>VorstellungID</u>, Datum, Uhrzeit, Verkauft, ↑<u>Saalnummer</u>, ↑<u>FilmID</u>)</p> <p>Gebucht(↑<u>Nummer</u>, ↑<u>VorstellungID</u>)</p> <p>Primärschlüssel sind unterstrichen, Fremdschlüsseln ist ein ↑ vorangestellt</p>                                                                                                                                                                                                                                                                                                                        | 2 | 2  |     |
| c1 | <p>Es handelt sich um einen Join der Tabellen Vorstellung und Film über das Attribut FilmID. Selektiert wird nach dem Datum 10.4.2004. Projiziert werden die Attribute Datum, Uhrzeit, Titel und FSK und zwar geordnet nach der Uhrzeit. Ausgabe des Filmprogramms für den 10.4.2004</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | 3 |    |     |
| c2 | <p>Geschachtelte Select-Anweisung. Die innere Select-Anweisung ermittelt mit einem Join über die FilmNr und einer Selektion die Action-Filme und bestimmt mittels MAX(Verkauft) den Action-Film mit größter Besucherzahl. Die äußere Select-Anweisung gibt dann einige Daten dieses Filmes aus. Bestimmung des Actionfilms mit größter Besucherzahl in einer Vorstellung.</p>                                                                                                                                                                                                                                                                                                                                                                                            |   | 3  |     |
| d1 | <pre>SELECT FilmID, Titel, Datum, Uhrzeit FROM Vorstellung, Film WHERE     Film.FilmID = Vorstellung.FilmID AND     Datum Between #1/1/504# AND #5/31/2004# AND     Kategorie='Abenteuer';</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 3 |    |     |
| d2 | <pre>SELECT VorstellungID, Datum, Uhrzeit</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |   | 3  |     |

| erwartete Schülerleistung |                                                                                                                                                                                                                                                                                                                                                            | I  | II | III |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|----|-----|
| d3                        | <pre> FROM Vorstellung, Saal, Film WHERE   Saal.Saalnummer = Vorstellung.Saalnummer AND   Vorstellung.FilmID = Film.FilmID AND   Titel = 'Herr der Ringe III' AND   Verkauft &lt; Sitzplätze  SELECT Film.FilmID, Sum(Verkauft) FROM Film, Vorstellung WHERE Film.FilmID=Vorstellung.FilmID GROUP BY Film.FilmID HAVING Sum(Verkauft) &gt; 1000 </pre>     |    |    | 3   |
| e                         | <p>Das ER-Diagramm muss um eine Tabelle Reservierung erweitert werden, die in einer n:1-Beziehung zur Tabelle Vorstellung steht.</p> <p>Technische Aspekte sind z. B. Transaktionsverwaltung, Datensicherheit und Web-Datenbanken</p> <p>Gesellschaftliche Aspekte sind z. B. Datenschutz und Vergleich zwischen telefonischer und online-Reservierung</p> |    | 3  | 3   |
| <b>Σ 35 BE</b>            |                                                                                                                                                                                                                                                                                                                                                            | 11 | 18 | 6   |

**ER-Diagramm zu Aufgabe a**



## C 1 Geldbeträge

### Aufgabenart

Diese Aufgabe behandelt die Themen Grammatiken und formale Sprachen sowie endliche Automaten und die Umsetzung eines Akzeptors mit der Programmiersprache Prolog.

### Voraussetzungen des Lehrplans

Zur Lösung der Aufgabe werden folgende verbindlichen Inhalte des Lehrplans benötigt: Formale Sprachen und Grammatiken samt Anwendung mit Syntaxdiagrammen, Zustandsdiagramm und Simulation eines endlichen Automaten, Kenntnisse in Prolog.

### Bewertungseinheiten

|   | erwartete Schülerleistung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | I | II | III |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|----|-----|
| a | <p>Das Syntaxdiagramm für Betrag lässt den Betrag 0 EUR und positive EUR-Beträge mit maximal drei Stellen zu. Beträge mit führenden Nullen sind ausgeschlossen. Es ist die Sprache der ganzzahligen Euro-Beträge von 0 EUR bis 999 EUR.</p>                                                                                                                                                                                                                                                                                                                                                                            | 3 | 3  |     |
| b | <p>Das Zustandsdiagramm sollte gemäß der Struktur der Syntaxdiagramme aufgebaut werden. Es sind nicht deterministische endliche Automaten mit oder ohne Epsilon-Übergängen möglich.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                | 2 | 4  |     |
| c | <p>Die Zustände des Automaten sind Variablen in der Grammatik. Jeder Übergang entspricht einer Produktion. Abkürzende Schreibweisen für die Behandlung der Ziffern 0 bis 9 sind erlaubt.</p> <p> <math>S \rightarrow 0V</math><br/> <math>S \rightarrow 1V \mid 2V \mid \dots \mid 9V</math><br/> <math>S \rightarrow 1P_1 \mid 2P_1 \mid \dots \mid 9P_1</math><br/> <math>S \rightarrow 1P_2 \mid 2P_2 \mid \dots \mid 9P_2</math><br/> <math>P_1 \rightarrow 0P_2 \mid 1P_2 \mid \dots \mid 9P_2</math><br/> <math>P_2 \rightarrow 0V \mid 1V \mid \dots \mid 9V</math><br/> <math>V \rightarrow " "W_1</math> </p> |   | 5  |     |

|   | <b>erwartete Schülerleistung</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | <b>I</b> | <b>II</b> | <b>III</b> |
|---|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|-----------|------------|
|   | $V \rightarrow , K_1$<br>$K_1 \rightarrow 0K_2 \mid 1K_2 \mid \dots \mid 9K_2$<br>$K_2 \rightarrow 0K_3 \mid 1K_3 \mid \dots \mid 9K_3$<br>$K_3 \rightarrow " "W_1$<br>$W_1 \rightarrow EW_2$<br>$W_2 \rightarrow UW_3$<br>$W_3 \rightarrow R$<br>Die Ableitung muss schrittweise stattfinden und der Bezug zur vorher erstellten Grammatik vorhanden sein.<br><br>$S \rightarrow 5P_1 \rightarrow 54P_2 \rightarrow 546V \rightarrow 546,K_1 \rightarrow 546,8K_2 \rightarrow 546,80K_3 \rightarrow 546,80 W_1 \rightarrow 546,80 EW_2 \rightarrow 546,80 EUW_3 \rightarrow 546,80 EUR$                                                         | 2        |           |            |
| d | Es handelt sich um einen Akzeptor für das Syntaxdiagramm Positiv. Die Zahlen 0 bis 999 werden akzeptiert. Die Bedeutung der Parameter Ein und Aus muss klar werden: der Parameter Ein ist die Eingabeliste, der Parameter Aus die Liste der noch nicht verarbeiteten Zeichen.<br>akzeptiere_zp akzeptiert obligatorisch eine der Ziffern 1 bis 9,<br>akzeptiere_zn akzeptiert optional eine der Ziffern 0 bis 9.                                                                                                                                                                                                                                 |          | 4         |            |
| e | Es ist ein vollständiger Akzeptor für Geldbeträge in Prolog zu entwickeln (wobei Aufgabe e zu berücksichtigen ist).<br>akzeptiere_betrag(Betrag):-<br>akzeptiere_null_positiv(Betrag, Rest),<br>akzeptiere_komma(Rest, Rest1),<br>akzeptiere_eur(Rest1).<br><br>akzeptiere_null_positiv(['0' R], R).<br>akzeptiere_null_positiv(L1, L2):-<br>akzeptiere_positiv(L1, L2).<br><br>akzeptiere_komma(L1, L2):-<br>L1 = [' R1],<br>akzeptiere_zk(R1, R2),<br>akzeptiere_zk(R2, L2).<br>akzeptiere_komma(L, L).<br><br>akzeptiere_zk([K R], R):-<br>member(K, ['0','1','2','3','4','5','6','7','8','9']).<br><br>akzeptiere_eur([' ', 'E', 'U', 'R']). | 2        |           | 5          |
|   | <b>Σ 30 BE</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 10       | 15        | 5          |