

# 46116 Frühjahr 2015

Softwaretechnologie / Datenbanksysteme (nicht vertieft)

Aufgabenstellungen mit Lösungsvorschlägen



**Die Bschlangaul-Sammlung**

Hermine Bschlangaul and Friends

# Aufgabenübersicht

Thema Nr. 1 . . . . .	3
Teilaufgabe Nr. 1 . . . . .	3
3. Projektmanagement [Gantt und CPM] . . . . .	3
Teilaufgabe Nr. 2 . . . . .	4
1. Relationale Anfragesprachen [Computer „Chiemsee“] . . . . .	4
Aufgabe 3 [Konsulat] . . . . .	6



## Die Bschlangaul-Sammlung

Hermine Bschlangaul and Friends

Eine freie Aufgabensammlung mit Lösungen von Studierenden für Studierende zur Vorbereitung auf die 1. Staatsexamensprüfungen des Lehramts Informatik in Bayern.



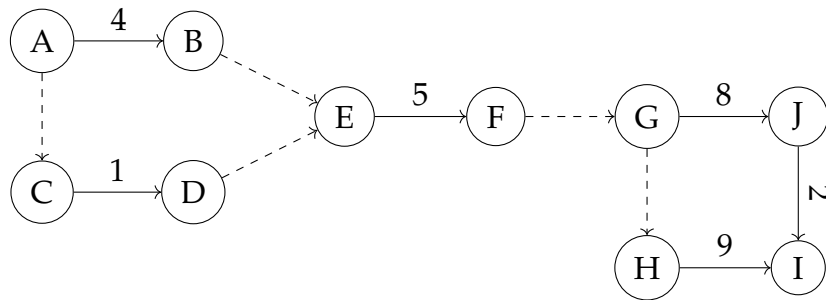
Diese Materialsammlung unterliegt den Bestimmungen der Creative Commons Namensnennung-Nicht kommerziell-Share Alike 4.0 International-Lizenz.

# Thema Nr. 1

## Teilaufgabe Nr. 1

### 3. Projektmanagement [Gantt und CPM]

Betrachten Sie folgendes CPM-Netzwerk:



- (a) Berechnen Sie die früheste Zeit für jedes Ereignis, wobei angenommen wird, dass das Projekt zum Zeitpunkt 0 startet.

Lösungsvorschlag

$i$	Nebenrechnung	$FZ_i$
A		0
B		4
C		0
D		1
E	$\max(4, 1)$	4
F		9
G		9
H		9
J		17
I	$\max(9_{(\rightarrow H)} + 9, 17_{(\rightarrow J)} + 2)$	19

- (b) Setzen Sie anschließend beim letzten Ereignis die späteste Zeit gleich der frühesten Zeit und berechnen Sie die spätesten Zeiten.

Lösungsvorschlag

--

$i$	Nebenrechnung	$SZ_i$
A	$\min(3, 0)$	0
B		4
C		3
D		4
E		4
F		9
G	$\min(10, 9)$	9
H		10
J		17
I		19

(c) Berechnen Sie nun für jedes Ereignis die Pufferzeiten.

Lösungsvorschlag

$i$	A	B	C	D	E	F	G	H	J	I
$FZ_i$	0	4	0	1	4	9	9	9	17	19
$SZ_i$	0	4	3	4	4	9	9	10	17	19
GP	0	0	3	3	0	0	0	1	0	0

(d) Bestimmen Sie den kritischen Pfad.

Lösungsvorschlag

$A \rightarrow B \rightarrow E \rightarrow F \rightarrow G \rightarrow J \rightarrow I$

(e) Was ist ein Gantt-Diagramm? Worin unterscheidet es sich vom CPM-Netzwerk?

Lösungsvorschlag

Diese Aufgabe hat noch keine Lösung. Hilf mit! Die Hermine schafft das nicht allein! Das ist ein Community-Projekt! Verbesserungsvorschläge, Fehlerkorrekturen, weitere Lösungen sind herzlich willkommen - egal wie - per Pull-Request oder per E-Mail an [hermine.bschlangaul@gmx.net](mailto:hermine.bschlangaul@gmx.net).

## Teilaufgabe Nr. 2

### 1. Relationale Anfragesprachen [Computer „Chiemsee“]

Gegeben sei folgendes relationales Schema, dessen Attribute nur atomare Attributwerte besitzen.

Computer: {IP, Name, Hersteller, Modell, Standort}

(a) Geben Sie für die folgenden Anfragen einen relationalen Ausdruck an:

(i) Geben Sie die IP-Adresse des Computers mit Namen „Chiemsee“ aus.

Lösungsvorschlag

$$\pi_{IP}(\sigma_{Name=Chiemsee}(Computer))$$

(ii) Geben Sie 2er-Tupel von IP-Adressen der Computer am selben Standort aus.

Lösungsvorschlag

$$\pi_{c1.IP, c2.IP}(\sigma_{c1.Standort=c2.Standort}(\rho_{c1}(Computer) \times \rho_{c2}(Computer)))$$

(b) Formulieren Sie die folgenden Anfragen in SQL:

(i) Geben Sie die IP-Adressen der Rechner am Standort „Büro2“ aus.

Lösungsvorschlag

```
SELECT IP FROM Computer WHERE Standort = 'Büro2';
```

(ii) Geben Sie alle Computer-Namen in aufsteigender Ordnung mit ihren IP-Adressen aus.

Lösungsvorschlag

```
SELECT Name, IP FROM Computer ORDER BY Name ASC;
```

(iii) Geben Sie für jeden Hersteller die Anzahl der unterschiedlichen Modelle aus.

Lösungsvorschlag

```
SELECT COUNT(DISTINCT Modell), Hersteller
FROM Computer
GROUP BY Hersteller;
```

(iv) Geben Sie für jeden Hersteller, welcher mindestens 2 unterschiedliche Modelle hat, die Anzahl der unterschiedlichen Modelle aus.

Lösungsvorschlag

```
SELECT Hersteller, COUNT(*) FROM Modelle GROUP BY Hersteller HAVING COUNT(*)
↪ > 1;
```

oder

Lösungsvorschlag

```
SELECT COUNT(DISTINCT Modell), Hersteller
FROM Computer
GROUP BY Hersteller
HAVING COUNT(DISTINCT Modell) >= 2;
```

```
-- AB 2 Einstieg Sql
```

```
-- Aufgabe 3: SQL-Anfragen auf einer Tabelle & Relationale Algebra
```

```
-- sudo mysql < Computer.sql
-- DROP DATABASE IF EXISTS Computer;
-- CREATE DATABASE Computer;
-- USE Computer;
```

```
CREATE TABLE Computer (
  IP VARCHAR(15) PRIMARY KEY NOT NULL,
  Name VARCHAR(30),
  Hersteller VARCHAR(30),
  Modell VARCHAR(30),
  Standort VARCHAR(30)
);
```

```
INSERT INTO Computer VALUES ('10.11.12.1', 'Chiemsee', 'HP', 'Spectre', 'Büro1');
INSERT INTO Computer VALUES ('10.11.12.2', 'Computer2', 'HP', 'Elite', 'Büro1');
INSERT INTO Computer VALUES ('10.11.12.3', 'Computer3', 'HP', 'Spectre', 'Büro1');
INSERT INTO Computer VALUES ('10.11.12.4', 'Computer4', 'HP', 'Elite', 'Büro1');
INSERT INTO Computer VALUES ('10.11.12.5', 'Computer5', 'HP', 'Spectre', 'Büro1');
INSERT INTO Computer VALUES ('10.11.12.6', 'Computer6', 'HP', 'Elite', 'Büro1');
INSERT INTO Computer VALUES ('10.11.12.7', 'Computer7', 'HP', 'Envy', 'Büro1');
INSERT INTO Computer VALUES ('10.11.12.8', 'Computer8', 'DELL', 'G3', 'Büro2');
INSERT INTO Computer VALUES ('10.11.12.9', 'Computer9', 'DELL', 'G7', 'Büro2');
INSERT INTO Computer VALUES ('10.11.12.10', 'Computer10', 'DELL', 'Latitude', 'Büro2');
INSERT INTO Computer VALUES ('10.11.12.11', 'Computer11', 'DELL', 'Alienware', 'Büro2');
INSERT INTO Computer VALUES ('10.11.12.12', 'Computer12', 'DELL', 'Inspiron', 'Büro2');
INSERT INTO Computer VALUES ('10.11.12.13', 'Computer13', 'DELL', 'XPS', 'Büro2');
INSERT INTO Computer VALUES ('10.11.12.14', 'Computer14', 'Apple', 'MacBook Air', 'Büro2');
INSERT INTO Computer VALUES ('10.11.12.15', 'Computer15', 'Apple', 'MacBook Air', 'Büro2');
INSERT INTO Computer VALUES ('10.11.12.16', 'Computer16', 'Apple', 'MacBook Air', 'Büro3');
INSERT INTO Computer VALUES ('10.11.12.17', 'Computer17', 'Apple', 'MacBook Air', 'Büro3');
INSERT INTO Computer VALUES ('10.11.12.18', 'Computer18', 'Apple', 'MacBook Air', 'Büro3');
INSERT INTO Computer VALUES ('10.11.12.19', 'Computer19', 'Apple', 'MacBook Air', 'Büro3');
INSERT INTO Computer VALUES ('10.11.12.20', 'Computer20', 'Apple', 'MacBook Air', 'Büro3');
INSERT INTO Computer VALUES ('10.11.12.21', 'Computer21', 'Apple', 'MacBook Air', 'Büro3');
INSERT INTO Computer VALUES ('10.11.12.22', 'Computer22', 'Apple', 'MacBook Air', 'Büro3');
INSERT INTO Computer VALUES ('10.11.12.23', 'Computer23', 'Apple', 'MacBook Air', 'Büro3');
```

### Aufgabe 3 [Konsulat]

Es sind folgende Informationen zu einer Datenbank für Konsulate gegeben:

- Jedes Konsulat hat einen Sitz in einer Stadt
  - Zu einem **Konsulat** soll ein eindeutiger *Name* (KonsulatName) (z. B. Konsulat Bayern), die *Adresse* und der *Vor-* (KVorname) bzw. *Nachname* (KNachname) des Konsuls gespeichert werden.
  - Für jede **Stadt** sollen der *Name* (StadtName), die *Anzahl der Einwohner* (EinwohnerAnzahl),
- ☐ E: Konsulat  
☐ A: Adresse  
☐ A: Vor-  
☐ A: Nachname  
☐ E: Stadt  
☐ A: Name  
☐ A: Anzahl der Einwohner

sowie das Land in dem es liegt, festgehalten werden. Gehen Sie davon aus, dass eine Stadt nur in Zusammenhang mit dem zugehörigen Land identifizierbar ist.

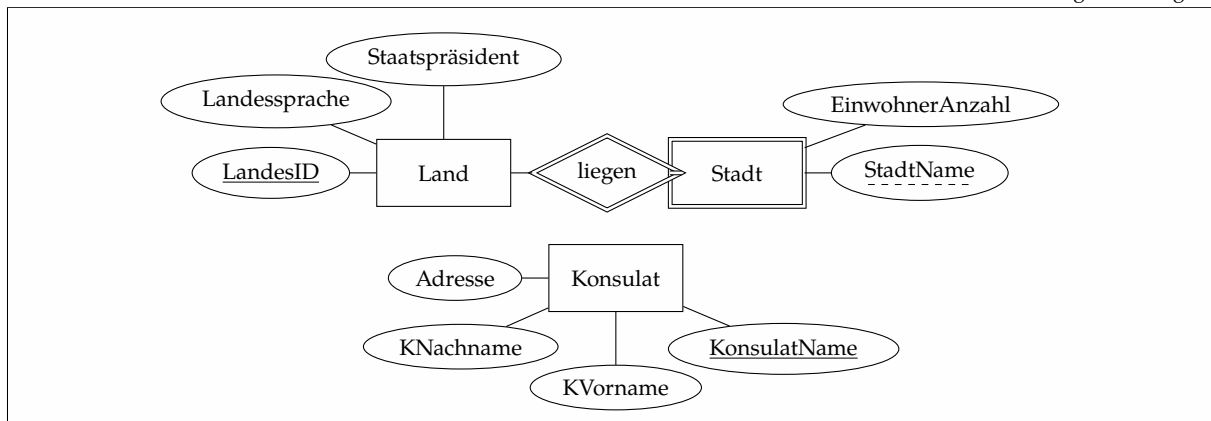
- Für ein **Land** soll der Name in *Landessprache*, der *Name des Staatspräsidenten* (Staatspräsident) und eine eindeutige *ID* (LandesID) gespeichert werden.

- (a) Entwerfen Sie für das obige Szenario ein ER-Diagramm in Chen-Notation. Bestimmen Sie hierzu:

- Die Entity-Typen, die Relationship-Typen und jeweils deren Attribute,
- Die Primärschlüssel der Entity-Typen, welche Sie anschließend in das ER-Diagramm eintragen, und
- Die Funktionalitäten der Relationship-Typen.

Hinweis: Achten Sie darauf, alle Totalitäten einzutragen.

Lösungsvorschlag



- (b) Überführen Sie das ER-Modell aus Aufgabe a) in ein verfeinertes relationales Modell. Geben Sie hierfür die verallgemeinerten Relationenschemata an. Achten Sie dabei insbesondere darauf, dass die Relationenschemata keine redundanten Attribute enthalten.

Lösungsvorschlag

Konsulat(KonsulatName, KVorname, KNachname, Adresse, StadtName, LandesID)

Stadt(LandesID, StadtName, EinwohnerAnzahl)

Land(LandesID, Landessprache, Staatspraesident)