



**Bestätigung der Verhaltensregeln**

Hiermit versichere ich, dass ich diese Klausur ausschließlich unter Verwendung der unten aufgeführten Hilfsmittel selbst löse und unter meinem Namen abgebe.

\_\_\_\_\_  
Unterschrift oder vollständiger Name, falls keine Stifteingabe verfügbar

## Einsatz und Realisierung von Datenbanksystemen

**Klausur:** IN2031 / Probe

**Datum:** Donnerstag, 23. Juli 2020

**Prüfer:** Prof. Dr. Alfons Kemper

**Uhrzeit:** 11:00 – 12:30

### Bearbeitungshinweise

- Diese Klausur umfasst **12 Seiten** mit insgesamt **11 Aufgaben**.  
Bitte kontrollieren Sie jetzt, dass Sie eine vollständige Angabe erhalten haben.
- Die Gesamtpunktzahl in dieser Prüfung beträgt 90 Punkte.
- Das Heraustrennen von Seiten aus der Prüfung ist untersagt.
- Als Hilfsmittel sind zugelassen:
  - ein **analoges Wörterbuch** Deutsch ↔ Muttersprache **ohne Anmerkungen**
- Mit \* gekennzeichnete Teilaufgaben sind ohne Kenntnis der Ergebnisse vorheriger Teilaufgaben lösbar.
- **Es werden nur solche Ergebnisse gewertet, bei denen der Lösungsweg erkennbar ist.** Auch Textaufgaben sind **grundsätzlich zu begründen**, sofern es in der jeweiligen Teilaufgabe nicht ausdrücklich anders vermerkt ist.
- Schreiben Sie weder mit roter / grüner Farbe noch mit Bleistift.
- Schalten Sie alle mitgeführten elektronischen Geräte vollständig aus, verstauen Sie diese in Ihrer Tasche und verschließen Sie diese.

Hörsaal verlassen von \_\_\_\_\_ bis \_\_\_\_\_ / Vorzeitige Abgabe um \_\_\_\_\_







### Aufgabe 3 IT-Sicherheit (6 Punkte)

Gegeben sei die Tabelle PrAnmeldung mit einigen Beispielwerten. Die Tabelle speichert für alle Studenten, zu welchen Prüfungen sie angemeldet sind.

PrAnmeldung		
MatrNr	Name	Pruefung
03642000	Thuy	NetSec
03654321	Alex	Modern DBs
03636363	Anna	Robotics
⋮	⋮	⋮

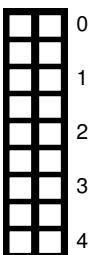
Es gibt eine Website, in der man sich nach Eingabe der Matrikelnummer alle seine Prüfungsanmeldungen auflisten lassen kann. Leider hat das Entwicklungsteam vergessen, die Benutzereingabe auf SQL-Injections zu prüfen.

Die im Formular genutzte Anfrage lautet:

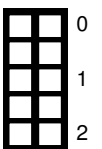
```
SELECT Pruefung FROM PrAnmeldung WHERE MatrNr={Benutzereingabe};
```

Schreiben Sie folgende SQL-Injections für das Feld {Benutzereingabe}:

a)\* Sie haben vergessen, sich rechtzeitig für ERDB anzumelden. Schreiben Sie eine SQL-Injection, um sich mit Ihrer Matrikelnummer (MatrNr) und Ihrem Namen (Name) zur Prüfung (Pruefung) „ERDB“ anzumelden.



b)\* Die Manipulation ist aufgefallen und Sie verwischen Ihre Spuren. Schreiben Sie eine SQL-Injection, um die gesamte Tabelle zu löschen.





## Aufgabe 4 Datalog (7 Punkte)

Gegeben seien die Fakten voraussetzen und vorlesungen aus der Universitätsdatenbank (hier nur in Ausschnitten dargestellt):

```
%%voraussetzen(VorgNr,NachfNr)
voraussetzen(5001,5041).
voraussetzen(5001,5043).
voraussetzen(5043,5022).
...
%%Vorlesungen(VorlNr, Titel, SWS, gelesenVon)
vorlesungen(5001,grundzuege,4,2137).
vorlesungen(5022,wissenschaftstheorie,3,2126).
vorlesungen(5041,ethik,4,2125).
vorlesungen(5043,erkenntnistheorie,3,2126).
...
```

Gegeben sei der folgende Ausdruck in Domänenkalkül:

$$\{[t] \mid \exists v,s,g([v,t,s,g] \in \text{Vorlesungen} \wedge \exists n([v,n] \in \text{voraussetzen} \wedge \exists s2,g2([n,\text{'Ethik'},s2,g2] \in \text{Vorlesungen})))\}$$

0		
1		

a)\* Nennen Sie den Zweck des Ausdrucks.

--

0			
1			
2			
3			

b)\* Übersetzen Sie den Ausdruck in Datalog.

--

0			
1			
2			
3			

c)\* Gegeben seien die folgenden Datalog-Prädikate:

```
vorgaenger(V,N) :- voraussetzen(V,N).
vorgaenger(V,N) :- vorgaenger(V,N),voraussetzen(V,N).
nichtVoraussetzen(V,NV) :- vorlesungen(V,_,_,_),vorlesungen(NV,_,_,_),
                           not(vorgaenger(V,NV)).
```

Begründen Sie stichpunktartig, ob das Prädikat nichtVoraussetzen stratifiziert ist.

--





## Aufgabe 5 Fragmentierung (8 Punkte)

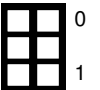
Gegeben sei folgende Relation *Klausur* mit Schlüssel *MatrNr*:

MatrNr	Name	Note	Standort
10101	Philipp	1,0	München
10102	Magdalena	1,0	Garching
10103	Erik	1,0	Garching
10104	Josef	1,0	Garching
10105	Alex	1,0	Garching
10106	Maxmilian	1,0	München

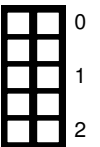
Für eine verteilte Datenbank soll die Tabelle geeignet fragmentiert werden. Ziel ist, Namen mit Standort der Studenten lokal und die Noten getrennt abzuspeichern.

Fragmentieren Sie die Relation geeignet *vertikal*.

a)\* Geben Sie das Schema für die zwei resultierenden Relationen *KlausurV<sub>1</sub>* und *KlausurV<sub>2</sub>* an.

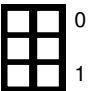


b) \* Geben Sie in SQL-92 die zwei resultierenden Relationen *KlausurV1* und *KlausurV2* als Hilfstabellen (mittels *with*) an.

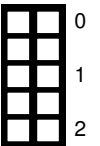


Die geeignetere der beiden resultierenden Relationen soll *horizontal* fragmentiert werden.

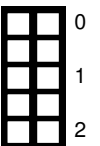
c)\* Geben Sie das Prädikat der Selektion an, mit dem fragmentiert wird.



d)\* Geben Sie in SQL-92 die zwei resultierenden Relationen *KlausurH1* und *KlausurH2* als Hilfstabellen (mittels *with*) an.



e) Schreiben Sie eine SQL-Abfrage, die die Ursprungsrelation aus den Teilrelationen zusammensetzt.





## Aufgabe 6 Window-Functions (10 Punkte)

Betrachten Sie die folgende Tabelle Waren mit verkauften Produkten in einem Supermarkt. Die Spalte verkauft besagt, wieviele Einheiten des jeweiligen Produktes verkauft worden sind.

Name	Preis	Kategorie	Verkauft
Brot	1.00	Backwaren	8128
Butter	0.80	Kühlwaren	496
Grill	60.00	Haushalt	6
Steak	8.00	Kühlwaren	28
...	...	...	...

0 ☐

1 ☐

2 ☐

3 ☐

a)\* Ermitteln Sie in SQL mittels Fensterfunktionen (Windowfunctions) den prozentualen Umsatzanteil jedes Produktes innerhalb seiner Kategorie.

0 ☐

1 ☐

2 ☐

3 ☐

b)\* Ermitteln Sie in SQL mittels Fensterfunktionen (Windowfunctions) für jedes Produkt das Mittel der Verkaufszahlen aus den 5 besser verkauften (höhere Verkaufszahlen) Produkten geordnet nach Verkaufszahlen.

0 ☐

1 ☐

2 ☐

3 ☐

4 ☐

c)\* Ermitteln Sie in SQL mittels Fensterfunktionen (Windowfunctions) die drei Produkte mit dem meisten Umsatz pro Kategorie.





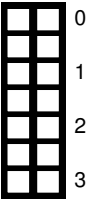
**Aufgabe 7 Skyline (6 Punkte)**

Gegeben sei die Relation Zuege:

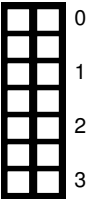
Baureihe	Gewicht	Vmax
401	795	280
402	455	280
403	459	330
406	488	330
407	495	320
411	402	230
412	670	250
415	311	230

Wir betrachten die Skyline über das Minimum des Attributs *Gewicht* und das Maximum über das Attribut *Vmax*.

a)\* Geben Sie die Anfrage, die die oben genannte Skyline in SQL-92 berechnet, an (d.h. ohne Skyline-Operator).



b)\* Geben Sie alle Tupel an, die in der Skyline enthalten sind. Es reicht, wenn Sie die zutreffenden Baureihen angeben.





## Aufgabe 8 Hauptspeicher-Datenbanken (9 Punkte)

Sie sollen für die Alexander-Maximilians-Universität (AMU) ein Hauptspeicherdatenbanksystem optimieren. In dem System sind die Daten aller Studenten gespeichert.

### Relationen

*Studenten*: MatrNr (8 Byte), Name (48 Byte), Studiengang (4 Byte), Semester (4 Byte)

MatrNr ist der Primärschlüssel der indiziert ist.

Schätzen Sie für jede der untenstehenden Anfragen einzeln, ob ein Row- oder Column-Store besser geeignet ist.

a) `* select * from Studenten;`

☐ Column-Store

☐ Row-Store

b) `* select Semester, count(*) from Studenten group by Semester;`

☐ Row-Store

☐ Column-Store

c) `* select Name, Studiengang, Semester from Studenten where MatrNr = 42;`

☐ Column-Store

☐ Row-Store

d) `* select Studiengang from Studenten where MatrNr = 42;`

☐ Row-Store

☐ Column-Store

e) `* select * from Studenten where Semester < 5;`

☐ Column-Store

☐ Row-Store

f) `* select * from Studenten where Semester = 25;`

☐ Column-Store

☐ Row-Store

g) `* insert into studenten values(4242, Max Meyer, Info, 7);`

☐ Column-Store

☐ Row-Store







## Aufgabe 9 XQuery (12 Punkte)

Als Grundlage für die Aufgabe wird eine leichte Abwandlung des Uni Schemas genutzt, von dem hier ein Ausschnitt geben ist. Fakultäten ist der Wurzelknoten des Dokuments unfak:

```
<Fakultaeten>
  <Fakultaet>
    <FakName>Theologie</FakName>
    <ProfessorIn PersNr="P2134">
      <Name>Augustinus</Name>
      <Rang>C3</Rang>
      <Raum>309</Raum>
      <Vorlesungen>
        <Vorlesung VorlNr="V5022">
          <Titel>Glaube und Wissen</Titel>
          <SWS>2</SWS>
        </Vorlesung>
        ...
      </Vorlesungen>
    </ProfessorIn>
  </Fakultaet>
  <Fakultaet>
    <FakName>Philosophie</FakName>
    <ProfessorIn ID="P2126" PersNr="P2126">
      <Name>Russel</Name>
      <Rang>C4</Rang>
      <Raum>232</Raum>
      <Vorlesungen>
        <Vorlesung ID="V5043" VorlNr="V5043" Voraussetzungen="V5001">
          <Titel>Erkenntnistheorie</Titel>
          <SWS>3</SWS>
        </Vorlesung>
        ...
      </Vorlesungen>
    </ProfessorIn>
  </Fakultaet>
</Fakultaeten>
```

a)\* Geben Sie in XPath die Namen der Professoren aus, die eine Vorlesung mit dem Titel Maeeutik halten.

0
1
2
3

b)\* Geben Sie in XPath alle Professoren aus, die mindestens zwei Vorlesungen halten.

0
1
2
3

c)\* Erstellen Sie in XQuery ein Vorlesungsverzeichnis geordnet nach Vorlesungstitel (aufsteigend) wie nachfolgend gezeigt.

```
<Vorlesungsverzeichnis>
  <Vorlesung Titel="Erkenntnistheorie"/>
  <Vorlesung Titel="Glaube und Wissen"/>
  ...
</Vorlesungsverzeichnis>
```

0
1
2
3
4
5
6





0			
1			
2			
3			
4			
5			
6			

a)\* Berechnen Sie für die folgenden drei Dokumente die TF-IDF Werte. Dabei sind alle Worte relevant.

1. ERDB macht echt viel Spaß
2. Die Klausur ist sicher machbar
3. Wir wünschen euch allen viel Erfolg bei der ERDB Klausur

--

0		
1		
2		
3		

b) Welches Ranking ergibt sich für die Anfrage: "ERDB Klausur"? Berechne Sie die Werte auf 3 Nachkommastellen genau.

Hilfswerte (gerundet):

$$\log(3) = 1/2$$

$$\log(2, 5) = 2/5$$

$$\log(2) = 1/3$$

$$\log(1, 5) = 1/6$$

$$\log(1) = 0$$





## Aufgabe 11 SPARQL (9 Punkte)

Im Folgenden ist schematisch eine RDF-Buchdatenbank dargestellt. Jedes Tripel (Subjekt, Prädikat, Objekt) enthält eine Information, z.B. Erscheinungsjahr oder Autorennamen, zu einem Buch. Jedes Märchen ist Teil eines Buches.

```
@prefix ex:<http://maerchen.example.org>.  
ex:KHG1 ex:hatAutor ex:Grimm.  
ex:KHG1 ex:erschienen 1812.  
ex:Rapunzel ex:teilVon ex:KHG1.  
ex:DerGestiefelteKater ex:teilVon ex:KHG1.  
ex:KHG2 ex:hatAutor ex:Grimm.  
ex:KHG2 ex:erschienen 1837.  
ex:SchneeweißchenUndRosenrot ex:teilVon ex:KHG2.  
ex:Kindermaerchen ex:hatAutor ex:Anderson.  
ex:Kindermaerchen ex:erschienen 1837.  
ex:KleineMeerjungfrau ex:teilVon ex:Kindermaerchen.
```

a)\* Werten Sie aus:

```
PREFIX ex:<http://maerchen.example.org>  
SELECT ?book  
WHERE {?book ex:hatAutor ex:Grimm. ?book ex:erschienen ?jahr FILTER (?jahr > 1836)}
```

--

0
1
2
3

b)\* Geben Sie jedes Märchen mit Erscheinungsjahr aus.

--

0
1
2
3

c)\* Geben Sie jeweils den Namen der Märchen aus Büchern von ex:Grimm mit einem Erscheinungsjahr nach 1836 aus.

--

0
1
2
3



This image shows a full page of blank graph paper. The grid consists of small, uniform squares formed by thin, light gray lines. There are no margins, text, or other markings on the page.