

Name:

Geburtsdatum:

Matrikelnummer:

Klausur
Informatische Werkzeuge in den Geistes-
und Sozialwissenschaften 2

25. Juli 2019

	Nur zur Korrektur, bitte freilassen!									
Aufgabe	1.1	1.2	2.1	3.1	3.2	4.1	5.1	6.1	Summe	Note
Möglich	4	5	20	3	8	10	4	6	60	
Erreicht										

Klausurnote:

Bonuspunkte:

Endnote:

Organisatorisches

Bitte lesen die folgenden Anweisungen genau und bestätigen Sie diese mit Ihrer Unterschrift.

1. Bitte legen Sie Ihren Studentenausweis und einen Lichtbildausweis zur Personenkontrolle bereit!
2. Die angegebene Punkteverteilung gilt unter Vorbehalt.
3. Es sind keine Hilfsmittel erlaubt außer eines handgeschriebenen "Spickzettels" von 1 Seite A4 einseitig.
4. Die Lösung einer Aufgabe muss auf den vorgesehenen freien Raum auf dem Aufgabenblatt geschrieben werden; die Rückseite des Blatts kann mitverwendet werden. Wenn der Platz nicht ausreicht, können bei der Aufsicht zusätzliche Blätter angefordert werden.
5. Wenn Sie die Prüfung aus gesundheitlichen Gründen abbrechen müssen, so muss Ihre Prüfungsunfähigkeit durch eine Untersuchung in der Universitätsklinik nachgewiesen werden. Melden Sie sich in jedem Fall bei der Aufsicht und lassen Sie sich das entsprechende Formular aushändigen.
6. Die Bearbeitungszeit beträgt 60 min.
7. Sie können 60 Punkte erreichen, wenn Sie alle Aufgaben vollständig lösen. Allerdings zählen 55 Punkte bereits als volle Punktzahl, d.h. 5 Punkte sind Bonuspunkte.
8. Überprüfen Sie Ihr Exemplar der Klausur auf Vollständigkeit (16 Seiten inklusive Deckblatt und Hinweise) und einwandfreies Druckbild! **Ver-gessen Sie nicht, auf dem Deckblatt die Angaben zur Person einzutragen und diese Erklärung zu unterschreiben!**

Erklärung: Durch meine Unterschrift bestätige ich den Empfang der vollständigen Klausurunterlagen und die Kenntnisnahme der obigen Informationen.

Erlangen, 25. Juli 2019

.....
(Unterschrift)

Bitte beachten Sie die folgenden Regeln, um keine Punkte zu verlieren:

- Wenn Sie eine Antwort auf einer anderen Seite fortsetzen, geben Sie bitte die Nummer der Aufgabe auf der neuen Seite mit an und verweisen Sie auf der alten Seite auf die neue.
- Begründen Sie Ihre Aussagen, wenn angebracht (wir würden gerne Teilpunkte für unvollständige Antworten geben). Wenn nicht explizit darum gebeten, antworten Sie möglichst nicht einfach mit „Ja“, „Nein“ oder „42“.

1 Versionskontrolle

Aufgabe 1.1 Bera Beispielperson führt in ihrer Shell folgende Befehle aus, um ein `git-Repository` zu klonen: 4 Pkt

```
git clone https://gitlab.cs.fau.de/iwgs-ss19/collaboration.git  
cd collaboration
```

Sie verändert danach lokal die Datei “`users.txt`”, indem sie einen Eintrag für sich selbst hinzufügt und führt folgende Befehle aus, um ihre lokalen Änderungen zu veröffentlichen:

```
git add users.txt  
git commit -m "Eintrag zu users.txt hinzugefuegt"  
git pull  
git push
```

Beschreiben Sie, warum Bera vor `git push` noch ein `git pull` ausgeführt hat und welches Problem auftreten könnte, wenn sie diesen Schritt überspringen würde.

Aufgabe 1.2 Erklären Sie das Konzept eines *merge conflicts*, unter welchen Umständen dieses Problem auftreten kann und wie es von einem Entwickelnden behoben werden kann. 5 Pkt

2 Datenbanken

Aufgabe 2.1 (Mitarbeitendenverwaltung)

Gegeben ist folgender Datensatz:

20 Pkt

PersonalID	Vorname	Nachname	Projekt	Projektstart
123	Erika	Musterfrau	Backend	2009
137	Jan	Modaal	Bildverarbeitung	2015
055	Ola	Nordmann	Bildverarbeitung	2015
249	Juan	Perez	Frontend	2010
005	Zhang	San	Backend	2009

Die **PersonalID** identifiziert einen Datensatz eindeutig (Primary Key). Sie wissen außerdem, dass für einen gleichen Eintrag in der Spalte **Projekt** auch der Eintrag in der Spalte **Projektstart** gleich sein wird.

1. **Tabelle Erstellen (5 Punkte)** Geben Sie die SQL-Befehle zum Erstellen einer Tabelle unter dem Namen **Employees** mit der obigen Struktur an. Sie müssen keine Einträge hinzufügen. Vermeiden Sie dabei eventuelle Probleme, falls bereits eine Tabelle namens **Employees** existiert.
2. **Daten anpassen (4 Punkte)**: Ola Nordmann wechselt von ihrem bisherigen Projekt (Bildverarbeitung) in das Backend-Projekt. Geben Sie den SQL-Ausdruck an, der den entsprechenden Datensatz anpasst.
3. **Redundanz (3 Punkte)**: In der obigen Tabelle werden einige Daten mehrfach gespeichert. Begründen Sie, warum das (gerade für größere Datensätze) suboptimal ist.
4. **Praxis (8 Punkte)**: Die obige Tabelle wurde in einer SQL-Datenbank `database.db` in einer Tabelle **Employees** gespeichert. Schreiben Sie ein `python`-Programm, welches diese Datenbank-Tabelle in ein Textdokument mit dem Namen `employees.txt` schreibt. Formatieren Sie das Dokument so, dass jeder Datensatz in eine neue Zeile geschrieben wird und die einzelnen Felder durch Kommas getrennt sind.

Hinweis: Mit einem `File`-Objekt `fl` können Sie die Methode `fl.writelines(strings)` verwenden, um eine Liste von Strings als Zeilen in eine Datei zu schreiben.

Diese Seite wurde für mehr Platz absichtlich leer gelassen.

Diese Seite wurde für mehr Platz absichtlich leer gelassen.

3 Bild

Aufgabe 3.1 (Kantenerkennung)

Beschreiben Sie kurz in eigenen Worten, wie Kantenerkennung auf Bildern funktioniert. 3 Pkt

Aufgabe 3.2 (Alpha-Blending)

Schreiben Sie eine python-Funktion `alphaBlend(bottomImage, topImage)`. Die beiden Argumente sind Bilder, repräsentiert als Listen von Listen (von Pixeln). 8 Pkt

- Die Einträge der inneren Listen sind alle jeweils ein Pixel, hier ein Tupel im Format (R,G,B,A).
- Die Pixelwerte in jedem Kanal sind normalisiert, also im Bereich zwischen 0 und 1.
- Ihre Funktion soll `topImage` über `bottomImage` legen, unter Beachtung des Alpha-Kanals.

Sie können für diese Aufgabe annehmen, dass `topImage` und `bottomImage` die gleichen Dimensionen haben.

Diese Seite wurde für mehr Platz absichtlich leer gelassen.

4 Web-Applikationen

Aufgabe 4.1 (Web-Applikation mit Liste)

Gegeben ist ein Bottle-Webserver. Dieser besteht aus zwei Dateien `Server.py` und `List.tpl`. 10 Pkt

Vervollständigen Sie folgendes Code-Gerüst, sodass auf der Webseite eine Überschrift und eine Liste angezeigt werden. Beide sollen in Abhängigkeit von den übergebenen Daten generiert werden: In der Überschrift soll der Wert der übergebenen Variable `heading` stehen und die Liste soll aus den Werten in `listData` bestehen.

```
# DATEI: Server.py
```

```
@bottle.route('/list')
```

```
def showList() :
```

```
    names = ['Michael', 'Jonas', 'Philipp']
```

```
    return bottle.template('List.tpl', heading='My list', listData=names)
```

```
<!-- Datei: List.tpl -->
```

```
<html>
```

```
    <head>
```

```
    </head>
```

```
    <body>
```

```
    </body>
```

```
</html>
```

Diese Seite wurde für mehr Platz absichtlich leer gelassen.

5 Geistiges Eigentum

Aufgabe 5.1 (CopyLeft)

Nennen Sie die „Copyleft“-Klausel in der GNU Public License oder in den Creative Commons Lizenzen, und erklären Sie kurz in eigenen Worten die Wirkungsweise. 4 Pkt

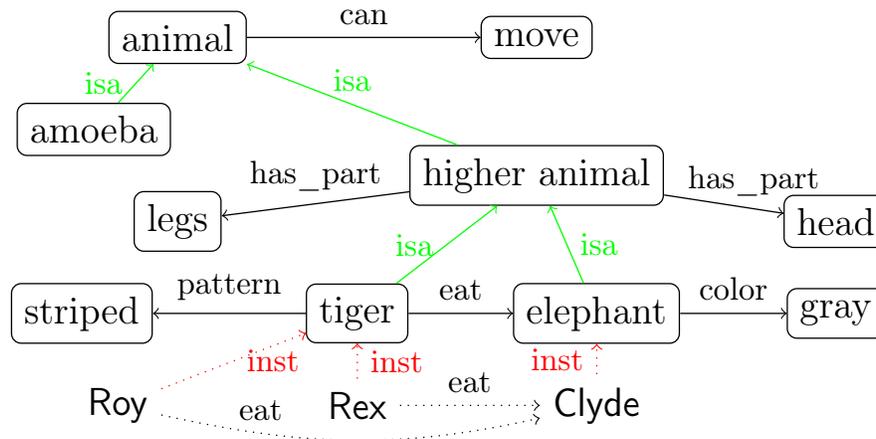
Diese Seite wurde für mehr Platz absichtlich leer gelassen.

6 Semantic Web

Aufgabe 6.1 (Semantische Netze)

Gegeben sei das folgende semantische Netz:

6 Pkt



1. Geben Sie je drei Subjekt/Prädikat/Objekt-Tripel aus der TBox und der ABox dieses Netzes an.
2. Zeichnen Sie jeweils drei Fakten und terminologische Axiome in das Netz ein, die im Netz inferiert werden können aber nicht explizit repräsentiert sind.

Diese Seite wurde für mehr Platz absichtlich leer gelassen.

Diese Seite wurde für mehr Platz absichtlich leer gelassen.

Diese Seite wurde für mehr Platz absichtlich leer gelassen.