Fachbereich für Computerwissenschaften

Prof. Dr. Nikolaus Augsten
Jakob-Haringer-Str. 2
5020 Salzburg, Austria
Telefon: +43 662 8044 6347
E-Mail: nikolaus.augsten@sbg.ac.at

UNIVERSITÄT SALZBURG

Datenbanken Vertiefung
Wintersemester 2019/2020

Prüfung
07.02.2020

Name: _____ **Matrikelnummer:** _____

Hinweise

- Bitte überprüfen Sie die Vollständigkeit des Prüfungsbogens (9 nummerierte Seiten).
- Schreiben Sie Ihren Namen und Ihre Matrikelnummer auf jedes Blatt des Prüfungsbogens und geben Sie alle Blätter ab.
- Grundsätzlich sollten Sie alle Antworten auf den Prüfungsbogen schreiben.
- Sollten Sie mehr Platz für eine Antwort benötigen, bitte einen klaren Verweis neben die Frage auf die Seitennummer des zusätzlichen Blattes setzen.
- Keinen Bleistift verwenden. Keinen roten Stift verwenden.
- Verwenden Sie die Notation und die Lösungsansätze, die während der Vorlesung besprochen wurden.
- Aufgaben mit mehr als einer Lösung werden nicht bewertet.
- Als Unterlage ist ein beliebig (auch beidseitig) beschriftetes A4-Blatt erlaubt.
- Zeit für die Prüfung: 90 Minuten

Unterschrift _____

Korrekturabschnitt — Bitte frei lassen

Aufgabe	1	2	3	4	5	6	7	8	Summe
Maximale Punkte	1	1	1	1	1	1	1	1	8
Erreichte Punkte									

Exercise 1 — 1 Point

Answer the following questions (be concise):

1. Explain three *undesirable phenomena* of concurrent transactions.
2. What does *read uncommited* stand for?
3. What is a *lost update* and how can it be prevented?

Name: Matrikelnummer:

Exercise 2 — 1 Point

Identify which of the properties: **conflict serializable**, **recoverable**, **cascadeless**, are fulfilled by the following schedule. If a property is not fulfilled, explain why.

T1:	T2:	T3:	T4:
read(A)			
write(A)			
		read(C)	
	write(B)		
	write(C)		
		read(A)	
			read(B)
read(A)			
	COMMIT		
		write(A)	
			write(B)
			COMMIT
read(B)			
COMMIT			
		COMMIT	

Exercise 3 — 1 Point

Consider the following schedule. Add 4 `COMMIT` instructions such that the resulting schedule is **not recoverable**. Can it still be cascadeless? Why (not)? Also, state if it is **conflict serializable** and give an equivalent serial schedule if possible.

T1:	T2:	T3:	T4:
	write(X)		
		read(X)	
read(X)			
	write(Y)		
read(Z)			
			read(Y)
			write(Z)

Name: Matrikelnummer:

Exercise 4 — 1 Point

Consider the following schedule and the **two-phase locking** scheduler.

T1:	T2:	T3:
read(X)		
	read(X)	
write(X)		
		read(X)
	write(X)	
		write(X)

Does the schedule result in a deadlock? Draw the wait–for graph. Assume TS(T1) < TS(T2) < TS(T3). Explain briefly the difference between wait–die and wound–wait deadlock prevention schemes on this example.

Exercise 5 — 1 Point

Can the following schedule be the output of a **strict two-phase locking** scheduler? If yes, add all required lock/unlock instructions. Otherwise, explain why.

```
T1:         T2:         T3:
_____________________________

read(X)

read(Y)

            read(X)

                        read(X)

                        write(X)

write(X)

            write(Y)

            COMMIT

                        read(X)

read(X)

                        COMMIT

COMMIT
_____________________________
```

Name: Matrikelnummer:

Exercise 6 — 1 Point

Consider the following schedule and the **validation based** scheduler. The timestamps correspond to the validation order, i.e., `TS(Ti)=validation(Ti)`.

T1:	T2:	T3:	T4:
	start		
start			
validate			
finish			
			start
		start	
	validate		
	finish		
		validate	
		finish	
			validate
			finish

The objects in the database that can be read or written are: A, B, C, D, E, F.

The read and write sets of the transactions are:

T1: R-set(T1)={A,C}, W-set(T1)={D,F}
T2: R-set(T2)={A,B}, W-set(T2)={C,D}
T3: R-set(T3)={C,E}, W-set(T3)={B,F}

Can `T3` be committed? Explain which conditions are satisfied or which one is violated.

Exercise 7 — 1 Point

Consider the following schedule. Indicate what happens when the schedule is processed by a **multiversion timestamp-ordering** scheduler. The transactions start in order with TS(T1)=1, TS(T2)=2, TS(T3)=3, TS(T4)=4. Assume that no versions of data item A exist.

T1:	T2:	T3:	T4:
write(A)			
	write(A)		
		write(A)	
	read(A)		
			read(A)
		write(A)	

Name: Matrikelnummer:

Exercise 8 — 1 Point

With the initial values A=100, B=200, C=300, write the log file for the following schedule. What happens during the recovery according to the recovery algorithm? Specify the resulting compensation log records.

T1:	T2:	T3:
START		
		START
	START	
		read(C)
	read(B)	
read(A)		
	B:=B-90	
A:=A-40		
		C:=C+70
		write(C)
write(A)		
	write(B)	
		COMMIT
----------CRASH----------		