Exercise Sheet 9: Transaction Processing

Exercise 1

The Phantom-Problem occurs if during the execution of a transaction T1 another transaction T2 adds data to the database and this data is not taken into account by T1.

a. Given are the following tables of a bank-application:

ACCOUNT

Kto-Nr	**Office**	**Balance**
227	Braunschweig	500
165	Hannover	2103
16	Hannover	1627

INVESTMENT

Office	**Sum**
Braunschweig	500
Hannover	3730

- A transaction T1 reads all accounts from the Hannover office and writes their balance (*Sum*) into the *investment* table.
- A transaction T2 creates a new account (Kto-Nr: 180; Balance: 750) and updates the *Sum* of the corresponding office in the *investment* table.

Show a schedule where the Phantom-Problem occurs.

b. How could this problem be solved using locks? Show a corresponding schedule!

Exercise 2

Given is the following schedule:

	T1	T2	T3	T4
1	read(A)			
2				read(A)
3	write(B)			
4				write(A)
5				commit
6		read(B)		
7			read(B)	
8	write(C)			
9	commit			
10		read(A)		
11		write(C)		
12		commit		
13			write(A)	
14			commit	

a. Show the corresponding conflict graph.

b. Is the schedule serializable? If yes, in which order would the transaction be executed in an equivalent serialized schedule?

c. Consider the scheduler uses the two-phase locking protocol. Show a corresponding schedule. If the scheduler can choose which operation to execute, it chooses the operation that has been occurred earlier in the input-schedule.