Note This example provides all the possible kinds of questions for the written exam. This version is actually longer than the written exam. In the written exam you will find three questions similar to those provided here.

Question I: SQL queries (6 pts.) Consider the relational schema in Table 1 where we have the following key constraints:

- The underlined attributes form the primary key of each relation.
- In `link` the attributes `document_name,author` are a foreign key to `hypertext`.
- In `link` the attribute `url` is foreign key to `webpage`.

hypertext					
document_name	lines	word_count	genre	year	author

link			
document_name	author	text	url

webpage		
url	visit_count	blacklisted

Table 1: Relational schema for Question I

Write the necessary queries in SQL to perform the following operations on the database:

1. Output the name of the hypertext, the author, and genre of all the hypertexts containing a link to a blacklisted page. (2 pts.)
2. For each genre with links to blacklisted pages, provide the total visits to black listed pages. Output the genre and the total visits. (2 pts.)
3. Output the name of the hypertext, the author, and genre of all the hypertexts without any link to a blacklisted page. (2 pts.)

Question II: Transactions (6 pts.)

1. Given the interleaved execution in Table 2 of operations:
   - Determine if and what conflicts arise from the concurrent execution of the transactions. (2 pts.)
   - Give a strict 2PL equivalent execution, and explain the locking order of variables. (2 pts.)
2. Given the interleaved execution in Table 3 draw the corresponding wait graph and determine if there are deadlocks and what transactions should be aborted in order to break it. (2 pts.)

T1	T2
R(A)	
R(C)	
W(C)	
	R(A)
	W(C)
Commit	
	Commit

Table 2: Execution of transactions for Question II (1)

T1	T2	T3
	R(B)	
R(A)		
		W(C)
		R(A)
		W(B)
		Commit
W(B)		
	R(A)	
	R(C)	
	W(C)	
	Commit	
W(A)		
Commit		

Table 3: Execution of transactions for Question II (2)

Question III: Graph databases (6 pts.) Consider the Entity-Relationship diagram in Figure 1: the entity sets are represented as rectangles, the relationship sets as diamonds, the attributes of each entity set are the attributes of the relations in Table 4, and the relationship set `sells` has the attribute `price`.

In the relational schema:

- In `sims` the attribute `name` is primary key.
- In `objects` the attribute `name` is primary key.
- In `interacts_with` the attributes `sim_name` and `object_name` are primary key.
- In `sells` the attribute `sim_name` and `object_name` are primary key.
- In `interacts_with` the attribute `sim_name` is foreign key to `name` in `sims` and `object_name` is foreign key to `name` in `objects`.
- In `sells` the attribute `sim_name` is foreign key to `name` in `sims` and `object_name` is foreign key to `name` in `objects`.

Answer the following questions:

1. Provide the create code for the database implementation in Neo4j. (3 pts.)
2. Implement the following queries 1 Exercise 1 in Neo4j for the given database implementation. (3 pts.)
   (a) Find the name of the sims and the type of the object they are interacting with. (1,5 pt.)
   (b) Find the total price of the objects sold by each sim. (1,5 pt.)

sims						
name	age	bladder	energy	happiness	position	money

objects		
name	type	position

interacts_with	
sim_name	*object_name*

sells		
sim_name	*object_name*	price

Table 4: Relational schema for Question I

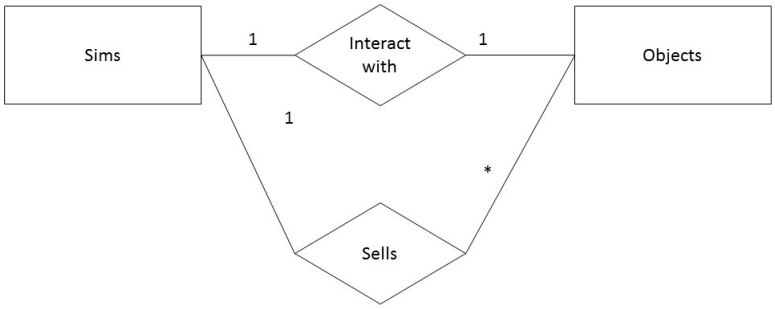

Figure 1: ERD for Exercise 3

Employee					
ssn	pnumber	hours	ename	pname	plocation

Table 5: Relational schema for Question IV

Question IV - Database normalization (6 pts.) Consider the relational schema in Table 5, with the following functional dependencies:

ssn → pnumber,hours
ssn → ename
pnumber → pname,plocation

- In what normal form is the table? Explain why by referring to the formal definition of the normal forms.(3 pts.)
- Use the algorithms for normalization to refine the relational schema. Use multiple refinements, i.e. from 1NF provides the steps to get a schema in 2NF, and from 2NF provide the steps to get a schema in BCNF.

Question V - Map-reduce (6pts.) Given the following SQL queries

```
select c.plate,c.model
from car c, owner o
where o.bsn = c.owner_bsn and
      c.model = 'Mercedes SLK'

select c.owner_bsn
from car c, owner o
where o.bsn = c.owner_bsn
having count(*) >= 2
```

provide an equivalent implementation using the map-reduce paradigm.