



**Instituut voor
Communicatie,
Media & Informatietechnologie**

Naam:

Studentnr:

Klas:

OPLEIDING : INFORMATICA

SOORT TENTAMEN : Tentamen

VOLTIJD / DEELTIJD : Voltijd

CURSUSCODE : INFDEV026B

CURSUSNAAM : Advanced databases and NoSQL

PERIODE : OP2

GROEPEN : INF2A, 2B, 2C, 2D, 2E, 2F en Herkansers

TIJDSDUUR IN MINUTEN : 150 minuten

CURSUSBEHEERDER : Ahmad Omar

AUTEUR : Francesco Di Giacomo

TWEEDE LEZER : Ahmad Omar

DIT TENTAMEN BESTAAT UIT :

VOORBLAD MET..... GENUMMERDE PAGINA'S

☐ .. MEERKEUZEVRAGEN

☐ .. CASUS

☒ .. OPEN VRAGEN

TOEGESTANE HULPMIDDELEN: pen, potlood, gum en Laptop.

SCHRIJF DUIDELIJK: wat niet duidelijk leesbaar is, wordt niet beoordeeld.

Schrijf je antwoorden en berekeningen:

☒ op het uitgereikte uitwerkingenpapier, dus niet op het tentamen
(vergeet niet naam en studentnummer op elk blad te vermelden)

☐ in het tentamen (zoals aangegeven)

☐ op het bijgeleverde antwoordformulier

BIJZONDERHEDEN:

Tentamen weer inleveren!

INFDEV026B Tentamen OP2 2017

Francesco Di Giacomo

Final grade (rounded): $0.4 * \text{Theory} + 0.6 * \text{Practice}$

Theory

Question I - Database normalization (5 pts.) Consider the relational schema in Table 1, with the following functional dependencies:

- $\text{company_code} \rightarrow \text{company_name}$
- $\text{flight_number} \rightarrow \text{departure, destination, distance, price}$
- $\text{departure, destination} \rightarrow \text{price}$

where the underlined attributes are the primary key.

license						
<u>flight_number</u>	<u>company_code</u>	departure	destination	distance	price	company_name

Table 1: Relational schema for Question I

- In what normal form is the table? Explain why by referring to the formal definition of the normal forms. (2 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. (3 pts.)

Question II: Transactions (5 pts.)

- Given the interleaved execution in Table 2 of operations:
 - Point out possible conflicts arising from the concurrent execution of the transactions. (1 pts.)
 - Give a strict 2PL equivalent execution, and explain the locking order of variables. (2 pts.)
- 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	T3
R(A)		
	R(B)	
		R(B) W(B)
R(C)	W(C)	
		R(A)
Commit		
	Commit	
		Commit

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

T1	T2	T3	T4
R(A)			
	R(B)		
		R(B) W(B)	
	W(A)		W(B)
W(B) Commit			
	Commit		
		Commit	
			Commit

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

Practice

Note: For this part you can use your laptop to test the code but you still need to copy the completed parts in the exam paper. You are not allowed to use any LINQ built-in functions for map-reduce (no **Select** or **Reduce** functions).

Question III: Map-Reduce (5 pts.) Consider relational database in Table 4, where **course_code** is a foreign key to the table **course** and **assignment_code** is a foreign key to the table **assignment**. Complete the missing pieces of code in the provided template to implement the SQL queries in the map-reduce paradigm. The notation **attribute: type** denotes the attribute name and its corresponding type.

Remember to copy the code you complete in the exam paper. Everything that is not in the exam paper will not be graded.

Todo 1: Complete the implementation of the **Map** function. (1 point)

Todo 2: Complete the implementation of the **Reduce** function. (1 point)

Todo 3: Complete the implementation of the first query below. (1 point)

Todo 4: Complete the implementation of the second query below. (1 point)

Todo 5: Complete the implementation of the third query below. (1 point)

Query 1:

```
SELECT name,duration
FROM courses
```

Query 2:

```
SELECT code,type
FROM assignment
WHERE month >= 9
```

Query3:

```
SELECT a.code, a.month, a.day
FROM course c, course_assignment ca, assignment a
WHERE c.code = ca.course_code AND a.code = ca.assignment_code
```

course		
<u>code</u> : string	name: string	duration: int

assignment			
<u>code</u> : string	month: int	day: int	type: string

course_assignment	
<u>course_code</u> : string	<u>assignment_code</u> : string

Table 4: Relational schema for Question III

Question IV: Graph databases (5 pts.) Consider the Entity-Relationship diagram in Figure 1: the entity sets are represented as rectangles, the relationship sets as diamonds, the underlined attributes are the primary key of each entity set. The duration of a course and an exam is expressed in hours. The chance of an exam is a positive number (1 for the first assessment, 2 for the retake, 3 for the first extra chance, and so on). Answer the following questions:

1. Provide the create code for the database implementation in Neo4j. (2 pts.)
2. Implement the following queries in Neo4j for the given database implementation. (3 pts.)
 - (a) Find all the exams given as second chance.
 - (b) Find all the assignments of the course DEV1 given before October.

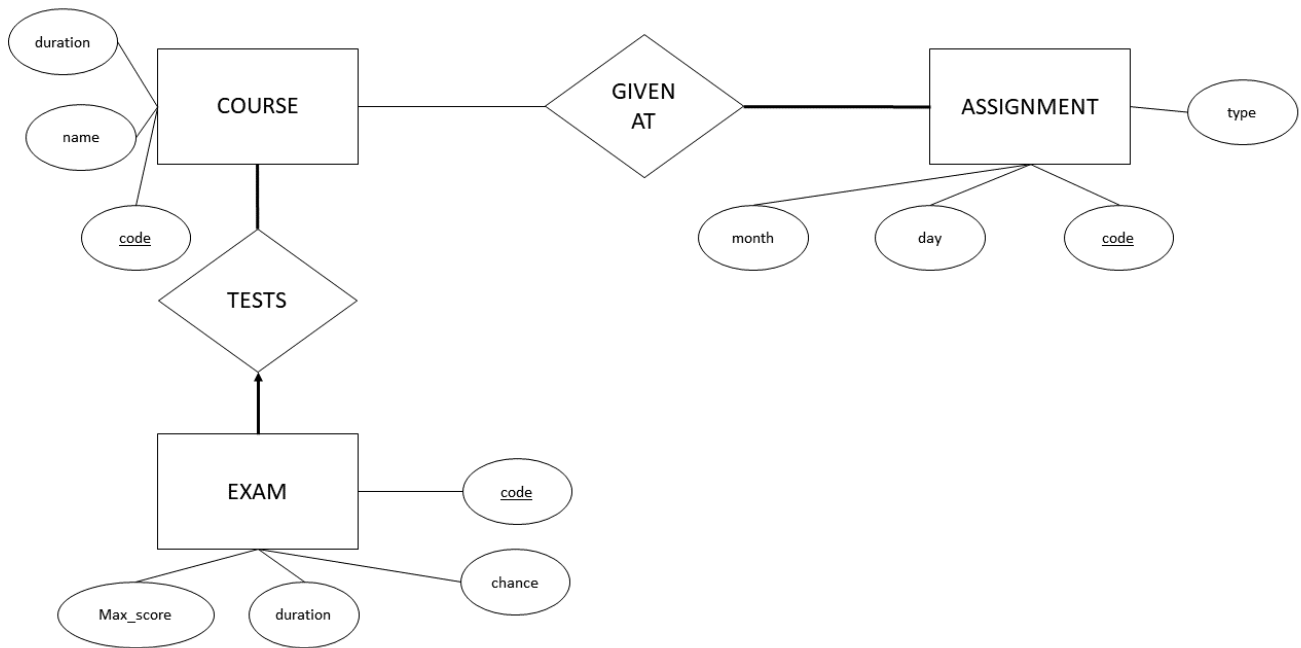


Figure 1: ERD for Question IV