MONASH University

Office Use Only

Semester Two 2021
Examination Period

Faculty of Information Technology

EXAM CODES: **FIT5137**

Sample Exam

TITLE OF PAPER: Advanced Database Technology - SAMPLE 2

EXAM DURATION: 2 hours 10 minutes or 130 minutes

THIS PAPER IS FOR STUDENTS STUDYING AT: (tick where applicable)

☐ Caulfield ☒ Clayton ☐ Parkville ☐ Peninsula
☐ Monash Extension ☐ Off Campus Learning ☐ Malaysia ☐ Sth Africa
☐ Other (specify)

During an exam, you must not have in your possession any item/material that has not been authorised for your exam. This includes books, notes, paper, electronic device/s, mobile phone, smart watch/device, calculator, pencil case, or writing on any part of your body. Any authorised items are listed below. Items/materials on your desk, chair, in your clothing or otherwise on your person will be deemed to be in your possession.

No examination materials are to be removed from the room. This includes retaining, copying, memorising or noting down content of exam material for personal use or to share with any other person by any means following your exam.
Failure to comply with the above instructions, or attempting to cheat or cheating in an exam is a discipline offence under Part 7 of the Monash University (Council) Regulations, or a breach of instructions under Part 3 of the Monash University (Academic Board) Regulations.

AUTHORISED MATERIALS

OPEN BOOK ☐ **YES** ☒ **NO**

CALCULATORS ☐ **YES** ☒ **NO**

SPECIFICALLY PERMITTED ITEMS ☐ **YES** ☒ **NO**
if yes, items permitted are:

Candidates must complete this section if required to write answers within this paper

STUDENT ID: __ __ __ __ __ __ __ __ DESK NUMBER: __ __ __ __ __

Part A	Part B.1	Part B.2	Part B.3	Part B.4	Part B.5	Total (60)

PART A – Multiple Choice Questions (Total 10 Marks)

Instructions:

- Part A is worth 10 marks.
- There are 10 questions in Part A and each question contains 1 mark.
- Choose one correct answer for each question.
- Attempt all questions in this part.

1. In MongoDB, write operations are atomic at the … level.
    a. Collection
    b. Document
    c. Row
    d. All of the above

2. A collection of related nodes in Cassandra is …
    a. Mem-table
    b. SSTable
    c. Cluster
    d. Data Center

3. Which of the following format is supported by MongoDB?
    a. XML
    b. JSON
    c. BSON
    d. SQL

4. What is going to be the result of `count` if we run the following query for the first time?

```
MERGE (n:Author {name:'Luke'})
ON CREATE SET n.count=1
ON MATCH SET n.count=count+1
RETURN n.count;
```

    a. 1
    b. 2
    c. count+1
    d. count+2

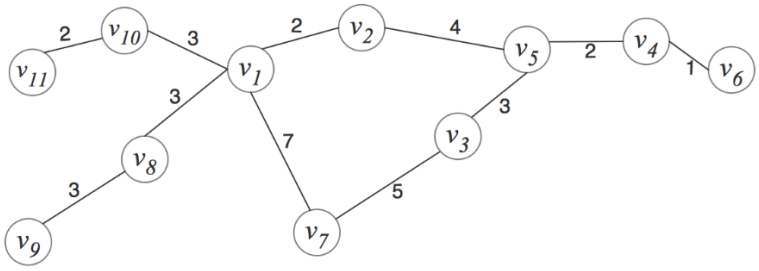
5. What does the following Cypher query express?

```
MATCH (person:Person)-[:KNOWS]-(friend:Person)-[:KNOWS]-
      (foaf:Person)
WHERE person.name = "Joe"
      AND NOT (person)-[:KNOWS]-(foaf)
RETURN foaf
```

   a. Find all of Joe's first-degree friends who do not have mutual friend with Joe.
   b. Find all of Joe's second-degree friends who do not have mutual friend with Joe.
   c. Find all of Joe's first-degree friends who are also not part of the second-degree friends.
   d. Find all of Joe's second-degree friends who are also not part of the first-degree friends.

6. A property of a vertex that indicates how far the vertex is from all others in the graph is …
   a. Closeness
   b. Betweenness
   c. Degree
   d. Order

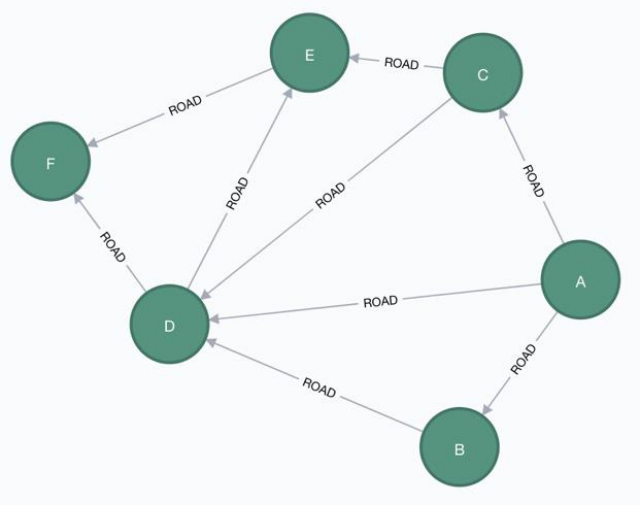
7. Given a graph as follow:



   Which of the following is **not** correct?
   a. Vertex v_1 has high betweenness as it forms a bottleneck in the graph.
   b. The shortest path from vertex v_3 to vertex v_8 is through vertex v_7 and vertex v_1.
   c. The graph is a weighted undirected graph.
   d. Vertex v_5 is a bottleneck.

8. Given the following graph and relationship data:

Loc Start	Loc Target	Road Cost
A	B	50
A	C	50
A	D	100
B	D	40
C	D	40
C	E	80
D	E	30
D	F	80



Which one of the following can correctly find a path with the least cost between nodes C and D?

a.
```
MATCH (start:Loc{name:'C'}), (end:Loc{name:'D'})
CALL algo.smallestPath(start, end, 'cost')
YIELD nodeId, cost
RETURN asNode(nodeId).name AS name, cost;
```

b.
```
MATCH (start:Loc{name:'C'}), (end:Loc{name:'D'})
CALL algo.smallestPath.stream(start, end, 'cost')
YIELD nodeId, cost
RETURN algo.asNode(nodeId).name AS name, cost;
```

c.
```
MATCH (start:Loc{name:'C'}), (end:Loc{name:'D'})
CALL algo.shortestPath.stream(start, end, 'cost')
YIELD nodeId, cost
RETURN algo.asNode(nodeId).name AS name, cost;
```

d.
```
MATCH (start:Loc{name:'C'}), (end:Loc{name:'D'})
CALL shortestPath (start, end, 'cost')
YIELD nodeId, cost
RETURN asNode(nodeId).name AS name, cost;
```

9. Which of the following is the correct way of updating a Set in Cassandra?

a.
```
UPDATE user
SET emails = emails + {'mary.mcdonald.AZ@gmail.com'}
WHERE first_name = 'Mary';
```

b.
```
UPDATE user
SET emails = emails + ['mary.mcdonald.AZ@gmail.com']
WHERE first_name = 'Mary';
```

c.
```
UPDATE user
SET emails = emails + <'mary.mcdonald.AZ@gmail.com'>
WHERE first_name = 'Mary';
```

d.
```
UPDATE user
SET emails[1] = 'mary.mcdonald.AZ@gmail.com'
WHERE first_name = 'Mary';
```

10. How do you list all databases in Mongo Shell?

a. db
b. use db
c. show db
d. show dbs

PART B – (Total 50 Marks)

Instructions:

- Part B is worth 50 marks.
- Attempt all questions in this part.
- All answers must be written in the answer section. The notes section will not be marked.

Question 1: = **Total 10 Marks**

You have been hired by MBooks to create a new database to record the information of their books and stocks. The amount of data they have so far is very little, which is shown in Table 1 and 2. Table 1 contains the details of the book which consist of the book title, author, and publish date. Table 2 provides the information of current stock/quantity and price of each book.

Table 1. Book details

ID	Title	Author	Publish Date
1	Introduction to SQL	Peter Shea	10/01/2018
2	NoSQL is fun	Sally Tucker	12/05/2012
3	I love NoSQL	Marcus Brown	10/10/2019

Table 2. Stock and price

ID	Title	Qty	Price
1	Introduction to SQL	110	$55
2	NoSQL is fun	52	$79.95
3	I love NoSQL	10	$60.50

MBooks have particularly asked you to use **Document-Oriented Database** to store their data due to the flexibility that Document-Oriented Database offers, especially when the database grows.

Questions:

a. Create and insert the data shown in the tables. **(3 marks)**

b. Answer the following queries **(5 marks)**:
   i. Show all books with quantity less than 20.
   ii. Find books published after 2018.
   iii. How much is the total price for two copies of "Introduction to SQL"?

c. Update the quantity of "I love NoSQL" to 15. **(1 mark)**

d. Add 5 more books for "NoSQL is fun". **(1 mark)**

Write your answers here:

a. **(3 marks)**

```
db.books.insertMany(
  [
    {
      "_id": 1,
      "title": "Introduction to SQL",
      "author": "Peter Shea",
      "publishDate": new Date("2018-01-10"),
      "qty": 110,
      "price":55
    },
    {
      "_id": 2,
      "title": "NoSQL is fun",
      "author": "Sally Tucker",
      "publishDate": new Date("2012-05-12"),
      "qty": 52,
      "price":79.95
    },
    {
      "_id": 3,
      "title": "I love NoSQL",
      "author": "Marcus Brown",
      "publishDate": new Date("2019-10-10"),
      "qty": 10,
      "price": 60.50
    }
  ]
)
```

Continue your answers here:

b. (5 marks)

i.
```
db.books.find(
  {"qty":{$lt:20}}
)
```
(1 mark)

ii.
```
db.books.find(
  {"publishDate":
     {$gte: ISODate("2019-01-01T00:00:00.000Z")}
  }
)
```
(1 mark)

iii.
```
db.books.aggregate(
  [
    { $match: {_id:1} },
    { $project: { title: 1,
                  total: {
                    $multiply: [ "$price", 2 ]
                  }
                }
    }
  ]
)
```
(Total 3 marks
1 mark for match, 1 mark for project,
1 mark for multiply)

Continue your answers here:

c. (1 mark)

```
db.books.update (
  {
    "_id": 3
  },
  {
    $set : {"quantity" : 15 }
  }
)
```

d. (1 mark)

```
db.books.update (
  {
    "_id": 2
  },
  {
    $inc: {"quantity" : 5 }
  }
)
```

+5 increment

Question 2: = **Total 10 Marks**

a. What is a Partitioner in Apache Cassandra and what is the benefit of having Partitioner Key? **(2 marks)**

b. List three types of partitioners are available in Cassandra and briefly explain each type. **(3 marks)**

c. What is Snitch? List and briefly explain two types of Snitches. **(4 marks)**

d. What is Tunable Consistency? **(1 mark)**

Write your answers here:

a. (2 marks)

- A partitioner determines how data is distributed across the nodes in the cluster.
- A partition key is used for data partitioning. Each row has a partition key that is used to identify the partition.

b. **(3 marks)**

Murmur3Partitioner – the default partitioner in Cassandra.
RandomPartitioner – similar to the Murmur3Partitioner except that it uses the MD5 (message-digest version 5) hash function to calculate the hash value.
ByteOrderedPartitioner – orders rows using partition key values.

Continue your answers here:

c. **(4 marks)**

- A snitch determines which data centers and racks nodes belong to. The job is a snitch is to determine relative host proximity for each node in a cluster. The snitch will figure out where nodes are in relation to other nodes.
- Two types of most popular snitches:
  - **Simple Snitch** - A simple snitch is used for single data centers with no racks.
  - **Property File Snitch** - A property file snitch is used for multiple data centers with multiple racks.

d. **(1 mark)**

Tunable Consistency is when the level of consistency can be specified as a trade-off with performance.

Question 3: = **Total 10 Marks**

a. What is CAP Theorem? Explain each component of CAP. **(3 marks)**

b. List some examples of database systems that are CA-based. **(2 marks)**

c. What does the BASE stand for? Explain each principle of BASE. **(3 marks)**

d. What is scaling up and scaling out? **(2 marks)**

Write your answer here:

a. **(3 marks)**

CAP Theorem states that it is impossible for a distributed system to provide following three guarantees at the same time:

- **Consistency**: All nodes see the same data, and the same versions of these data, at the same time.
- **Availability**: Every request receives a response indicating a success or failure result.
- **Partition Tolerance**: The system continues to work even if nodes go down or are added. The distributed system can cope with it being divided into two or more disjoint network partitions due to node or network failure.

b. **(2 marks)**

The traditional relational databases are usually CA-based. Some example database systems: Oracle SQL, MySQL, PostgreSQL, Microsoft SQL Server.

Continue your answers here:

c. (3 marks)

BASE stands for Basically Available, Soft state and Eventual consistency.

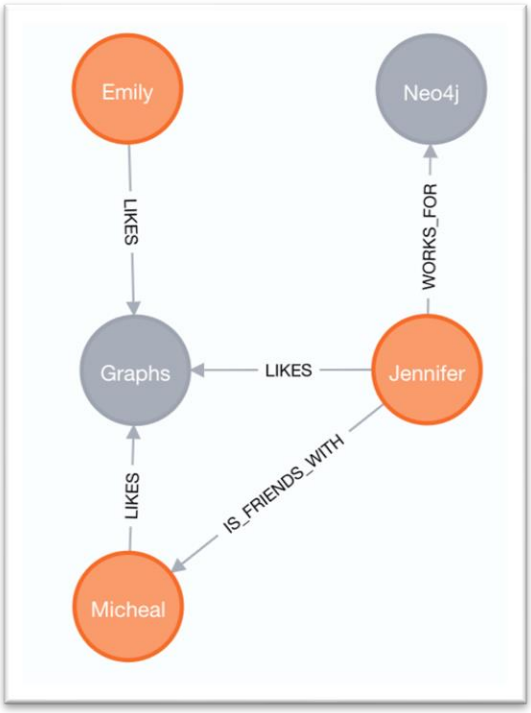
- **Basically Available**: Measures are in place to guarantee availability under all circumstances, if necessary, at the cost of consistency.
- **Soft state**: The state of the database may evolve, even without external input, due to the asynchronous propagation of updates throughout the system.
- **Eventual Consistency**: The database will become consistent over time, but may not be consistent at any moment and especially not at transaction commit.

d. (2 marks)

- **Scaling up:**
  Scaling entails upgrading an existing database server to add additional processors, memory, network bandwidth, or other resources that would improve performance on a database management system. Scaling up involves migrating and improving system to a larger system while maintaining the same number of systems. It could also entail replacing an existing server with one that has more CPUs, memory, and so forth.
- **Scaling out:**
  Scaling out entails adding servers to a cluster. Scaling out involves spreading out workloads across several servers when then workloads surpass the capacity of the current server.

Question 4: = **Total 10 Marks**

Given the following graph that contains three person nodes (i.e. Emily, Jennifer and Micheal), one technology node (i.e. Graph) and one company node (i.e. Neo4j).



Questions:

a. Write the code to create the above graph in a single run. **(4 marks)**

b. Suppose that the above graph is only a portion of a larger database. Write the code for finding the names of people who like Graphs, SQL, or Python. **(2 marks)**

c. If each person node were to have the properties name. phone number, email, and address as follow:

   Name: Jennifer
   Phone number: 0422222677
   Email: jenny@gmail.com
   Address: 123 Station Rd, Clayton, VIC

   Redraw the graph to optimize querying for the people who have Clayton suburb in Victoria in their address. (Assumption: Clayton suburb may also exist in other states within Australia). Please note: you are not required to provide any code for this sub-question. **(4 marks)**

Write your answers here:

a. **(4 marks)**

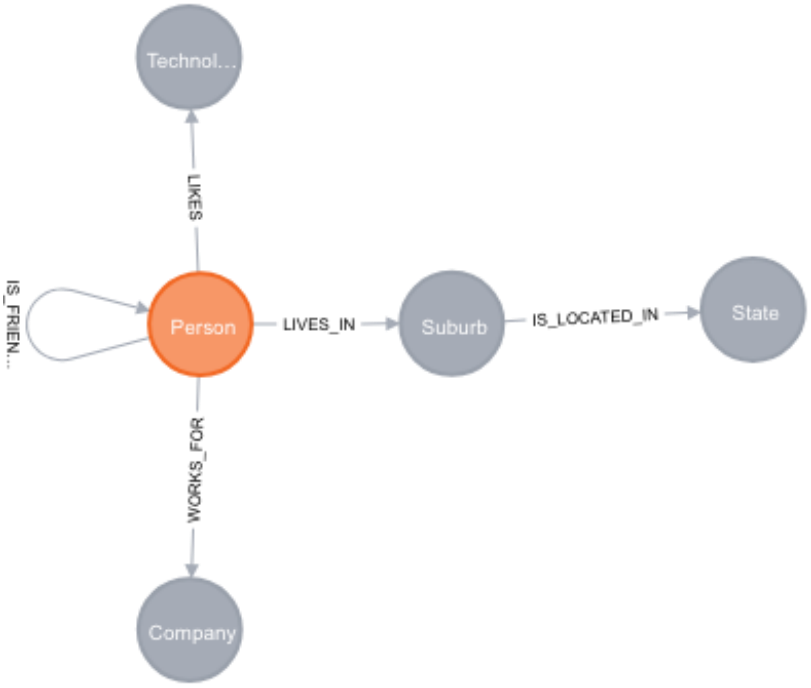
```
MERGE (e:Person{name:'Emily'})
MERGE (j:Person{name:'Jennifer'})
MERGE (m:Person{name:'Micheal'})
MERGE (g:Technology {type:'Graph'})
MERGE (n:Company {name:'Neo4j'})


MERGE (e)-[:LIKES] -> (g)
MERGE (m)-[:LIKES] -> (g)
MERGE (j)-[:LIKES] -> (g)
MERGE (j)-[:WORKS_FOR] -> (n)
MERGE (j)-[:IS_FRIENDS_WITH] -> (m)
```

b. **(2 marks)**

```
MATCH (p:Person)
MATCH (p)-[:LIKES]->(t:Technology)
WHERE t.type IN ['Graph', 'SQL', 'Python']
RETURN p.name
```

Continue your answers here:

c. (4 marks)



Question 5: = **Total 10 Marks**

a. Describe two reasons for choosing a graph database for your application. **(2 marks)**

b. Name two use cases for graph databases. **(2 marks)**

c. Is there any join operation in graph databases? If yes, explain how the join operation works. If no, how do graph databases avoid joins? **(2 marks)**

d. How is the MATCH statement like a SQL SELECT statement? **(1 mark)**

e. Explain why graph databases tend to struggle with scaling out. **(3 marks)**

Write your answers here:

a. (2 marks)

why use graph database

- There is a need to model explicit relations between entities and rapidly traverse paths between entities.
- There is an affinity between the problem domain, such as transportation networks and graphs.

b. (2 marks)

- Modeling computer networks
- Modeling social media networks

Continue your answers here:

c. (2 marks)

no join graph database

There is **no** join in graph database. In a graph database, instead of performing joins, you follow edges from vertex to vertex. The relationships are represented by the edges in which we can traverse through the edges to find the connection between entities.

d. (1 marks)

MATCH is used to retrieve data from a graph database. MATCH supports filtering based on properties.

e. (3 marks)

Graph databases are designed to store data with complex relationships. The data in a graph database are tightly integrated to one another. Scaling out involves moving data across several servers, which means that the data needs to be relatively independent in order to be spread across different servers. Because of the highly related data nature of graph databases, scaling out will cause communication overhead. Therefore, graph databases tend to perform best in centralized or lightly clustered environments, instead of a scaled-out environment.

Continue your answers here:

THE END