UNIVERSITY OF LONDON

CM3010

BSc EXAMINATION

COMPUTER SCIENCE

Databases and Advanced Data Techniques

Release date: Tuesday 19 March 2024 at 12:00 midday Greenwich Mean Time

Submission date: Wednesday 20 March 2024 by 12:00 midday Greenwich Mean Time

Time allowed: 4 hours to submit

INSTRUCTIONS TO CANDIDATES:
Part A of this assessment consists of a set of **TEN** Multiple Choice Questions (MCQs). You should attempt to answer **ALL** the questions in **Part A.** The maximum mark for Part A is **40**.

Candidates must answer **TWO** out of the **THREE** questions in **Part B**. The maximum mark for Part B is **60**.

Part A and Part B will be completed online together on the Inspera exam platform. You may choose to access either part first upon entering the test area but must complete both parts within **4 hours** of doing so.

A handheld non-programmable calculator may be used when answering questions on this paper but it must not be able to display graphics, text or algebraic equations. Please hold your calculator to the camera at the start of the examination to clearly show the make and type.

You may use **ONE** A4 page of previously prepared notes in this examination. Please hold up your notes to the camera at the start of the examination.

File upload is **NOT** permitted.

Do not write your name anywhere in your answers.

PART A

Candidates should answer the **TEN** Multiple Choice Questions (MCQs) in Part A.

PART B

Candidates should answer any **TWO** questions from Part B.

Question 2

```
@prefix schema: <http://schema.org/> .
@prefix gnd: <http://d-nb.info/standards/elementset/gnd#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix chm: <http://data.carnegiehall.org/model/> .
@prefix chi: <http://data.carnegiehall.org/instruments/> .
@prefix wd: <http://www.wikidata.org/entity/> .
@prefix wdt: <http://www.wikidata.org/prop/direct/> .

<http://data.carnegiehall.org/names/18065> a chm:Entity, schema:Person ;
    rdfs:label "Maria Callas" ;
    gnd:playedInstrument chi:61 ;
    schema:birthDate "1923-12-02"^^xsd:date ;
    schema:birthPlace <http://sws.geonames.org/5128581/> ;
    schema:deathDate "1977-09-16"^^xsd:date ;
    schema:name "Maria Callas" ;
    skos:exactMatch <http://dbpedia.org/resource/Maria_Callas>,
        <http://id.loc.gov/authorities/names/n50032183>,
        wd:Q128297,
        <https://musicbrainz.org/artist/9dee40b2-25ad-404c-9c9a-139feffd4b57> .
```

(a) The above text is from the Carnegie Hall data lab. You can request their data in various RDF serialisations:

i. Which RDF serialisation is this?

[1 mark]

ii. Name **ONE** other serialisation and, briefly, describe the difference.

[2 marks]

iii. How many triples are shown here?

[1 mark]

(b) Following two of the URLs in the above – chi:61 and wd:Q128297 gives the following extra triples:

```
chi:61 a <http://purl.org/ontology/mo/Instrument> ;
    rdfs:label "soprano" .

wd:Q128297 wdt:P1477 "Maria Anna Cecilia Sofia Kalogeropoulou"@en,
             "Μαρία Άννα Καικιλία Σοφία Καλογεροπούλου"@el .

wd:P1477 schema:description "full name of a person at birth, if different from
their current, generally used name"@en .
```

i. What is the full URL of wd:Q128297?

[1 mark]

ii. Given a triplestore with the RDF from these resources and a SPARQL endpoint, what query would list the birth name of all Sopranos?

[5 marks]

iii. Both Wikidata and Carnegie Hall have SPARQL endpoints, but the Carnegie Hall triplestore does not include Wikidata's triples, and Wikidata does not have Carnegie Hall data. Give **TWO** ways that queries like the one you give in (ii) could still be carried out.

[5 marks]

(c) Your project wants to use biographical data from Wikidata, concert listings from Carnegie Hall, and MusicBrainz discographies. Consider the relative merits and practicality of using the **THREE** existing resources as live Linked Open Data as opposed to downloading the data from each and creating a relational database for the data you need.

[9 marks]

(d) Wikidata uses almost exclusively their own ontology with a bespoke set of properties and classes. Carnegie Hall Data Labs primarily use ontologies from other projects, especially schema.org. Why might they have chosen different approaches? What are the benefits of each?

[6 marks]

Question 3

The UK Government issues data on exam attainment for 16–18 year-olds in CSV files. The following is an extract from the data (rotated to fit here):

Characteristic type	Gender	Gender	All students	Free School Meals
Characteristic	Male	Female	State-funded students	Eligible for FSM
Subject name	Additional Mathematics	Classical Greek	Textiles Technology	Total STEM subjects
Subject Area	Maths	Classical Studies	Design and Technology	All STEM subjects
Total Students	z	100	661	6084
Total Students all subjects	z	145989	228782	14865
Number at grade A*	z	27	27	372
Number at grade A	z	54	85	932
Number at grade B	z	9	164	1067
Number at grade C	z	9	199	1204
Number at grade D	z	0	115	1231
Number at grade E	z	1	58	847
Number at grade U	z	0	13	431
Number achieving grade A*-A	z	81	112	1304
Number achieving grade A*-B	z	90	276	2371
Number achieving grade A*-C	z	99	475	3575
Number achieving grade A*-D	z	99	590	4806
Number achieving grade A*-E	z	100	648	5653
Percent achieving grade A*	z	27	4.08	6.1144

(a) What Normal Forms (if any) is this table in? Justify your answer.

[2 marks]

(b) The CSV uses “Z” to indicate “not applicable”. What problems might this create for SQL implementations? How would you avoid them?

[3 marks]

(c) Design a relational model for the files, and give the CREATE commands needed. Explain your choices and show what Normal Forms your solution is in.

[15 marks]

(d) Give a query for your database that retrieves the percentage of A*-C grades for Classical Studies for each 'Characteristic' that the files track.

[4 marks]

(e) Is a relational model the best approach for this sort of data? Evaluate (briefly) this approach and at least two alternative models.

[6 marks]

Question 4

The MongoDB website gives the following as an example of a JSON document for a document database:

```
{
     "_id": 1,
     "first_name": "Tom",
     "email": "tom@example.com",
     "cell": "765-555-5555",
     "likes": [
        "fashion",
        "spas",
        "shopping"
     ],
     "businesses": [
        {
           "name": "Entertainment 1080",
           "partner": "Jean",
           "status": "Bankrupt",
           "date_founded": {
              "$date": "2012-05-19T04:00:00Z"
           }
        },
        {
           "name": "Swag for Tweens",
           "date_founded": {
              "$date": "2012-11-01T04:00:00Z"
           }
        }
     ]
  }
```

(a) Given a database of documents like this:

i. What query would return documents for people who like spas?

[2 marks]

ii. What query would find individuals with businesses founded before the first of March, 2020, who also have at least one business with the status of "Bankrupt"?

[4 marks]

(b) A bug in the data entry form for this database created several records with "likes” including "fashun" rather than "fashion".

i. How would you construct a query that would fix entries with the wrong data? (Explain in words – you do not need to know the full syntax).

[4 marks]

ii. A colleague argues that this is a problem of referential integrity, and that you would be able to avoid this issue using a Linked Data database or a relational database. In each case, what strategy would you use?

[4 marks]

iii. List all the tables you would need for a relational model of this data, including primary and foreign keys for each.

[8 marks]

iv. Evaluate these **THREE** models (document, relational and Linked Data/graph) for this sort of data. What would you need to know about the intended application to decide between them?

[8 marks]

END OF PAPER