



BSc EXAMINATION

COMPUTER SCIENCE

Databases and Advanced Data Techniques

Release date: Monday 6 March 2023 at 12:00 midday Greenwich Mean Time

Submission date: Tuesday 7 March 2023 by 12:00 midday Greenwich Mean Time

Time allowed: 24 hours to submit

INSTRUCTIONS TO CANDIDATES:

Section A of this assessment paper consists of a set of **TEN** Multiple Choice Questions (MCQs) which you will take separately from this paper. You should attempt to answer **ALL** the questions in Section A. The maximum mark for Section A is **40**.

Section A will be completed online on the VLE. You may choose to access the MCQs at any time following the release of the paper, but once you have accessed the MCQs you must submit your answers before the deadline or within **4 hours** of starting whichever occurs first.

Section B of this assessment paper is an online assessment to be completed within the same 24-hour window as Section A. We anticipate that approximately **1 hour** is sufficient for you to answer Section B. Candidates must answer **TWO** out of the **THREE** questions in Section B. The maximum mark for Section B is **60**.

Calculators are not permitted in this examination. Credit will only be given if all workings are shown.

You should complete **Section B** of this paper and submit your answers as **one document**, if possible, in Microsoft Word or a PDF to the appropriate area on the VLE. You are permitted to upload 30 documents. However, we advise you to upload as few documents as possible. Each file uploaded must be accompanied by a coversheet containing your **candidate number**. In addition, your answers must have your candidate number written clearly at the top of the page before you upload your work. Do not write your name anywhere in your answers.

SECTION A

Candidates should answer the **TEN** Multiple Choice Questions (MCQs) quiz, **Question 1** in Section A on the VLE.

SECTION B

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

Question 2

OpenDocument files are zipped bundles of resources, including copies of images, metadata and content. Here is an extract from the content file from a .odt file.

```
<office:text>
  <text:sequence-decls>
    <text:sequence-decl text:display-outline-level="0"
                        text:name="Illustration"/>
    <text:sequence-decl text:display-outline-level="0"
                        text:name="Table"/>
    <text:sequence-decl text:display-outline-level="0"
                        text:name="Text"/>
    <text:sequence-decl text:display-outline-level="0"
                        text:name="Drawing"/>
  </text:sequence-decls>
  <text:p text:style-name="Standard">Consider the following three
    data structures:</text:p>
  <text:list xml:id="list7123963207625859" text:style-name="L1">
    <text:list-item>
      <text:p text:style-name="P1">Trees</text:p>
    </text:list-item>
    <text:list-item>
      <text:p text:style-name="P1">Graphs</text:p>
    </text:list-item>
    <text:list-item>
      <text:p text:style-name="P1">Relations</text:p>
    </text:list-item>
  </text:list>
  <text:p text:style-name="Standard">Each has its own special
    advantages, and each is capable of representing the others
    where necessary.</text:p>
</office:text>
```

- (a) What language is this encoded in? [1]
- (b) What data structure does it use? [1]

(c) List the two namespaces that this document uses. [2]

(d) What would the xpath expression `//text:list-item/text:p` return? Would it be different to `//text:list//text:p`? [4]

(e) The code below is from the OpenDocument RelaxNG schema

```
<define name="text-list">
  <element name="text:list">
    <ref name="text-list-attr"/>
    <optional>
      <ref name="text-list-header"/>
    </optional>
    <zeroOrMore>
      <ref name="text-list-item"/>
    </zeroOrMore>
  </element>
</define>
<define name="text-list-header">
  <element name="text:list-header">
    <ref name="text-list-item-content"/>
  </element>
</define>
```

How does this code help us assess if the document above is well-formed? [2]

(f) How does this code help us assess if the document above is valid? [2]

(g) Which part or parts of the document is this relevant to? [2]

(h) Give an example of an element that would not be valid given this schema code (assume that `text-list-attr` only defines attributes). [3]

(i) Assess the suitability of this data structure for encoding word processing documents. What advantages or disadvantages would a relational model bring? [13]

Question 3

MusicBrainz is an online database of information about recordings. The following is returned when I make a request for `https://musicbrainz.org/artist/0d79fe8e-ba27-4859-bb8c-2f255f346853` with a special accept-type in the request header.

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix schema: <http://schema.org/> .
```

```
<http://musicbrainz.org/artist/0d79fe8e-ba27-4859-bb8c-2f255f346853>
  schema:foundingDate "2013-06-13"^^schema:Date ;
  schema:member [
    schema:member <http://musicbrainz.org/artist/09720eec-\
3871-49d5-932d-eb7542768cd3> ;
    schema:startDate "2013-06-13"^^schema:Date ;
    rdf:type schema:OrganizationRole
  ] ,
  [
    schema:member <http://musicbrainz.org/artist/23c8056b-\
ee13-4cfc-a772-2f5292e35bb5> ;
    schema:startDate "2013-06-13"^^schema:Date ;
    rdf:type schema:OrganizationRole
  ] ,
  ...
  schema:name "BTS" ;
  schema:sameAs <http://bts-official.jp/> .

<http://musicbrainz.org/artist/23c8056b-ee13-4cfc-a772-2f5292e35bb5>
  schema:name "방탄소년단0...방탄소년단" ;
  rdf:type schema:MusicGroup , schema:Person .

<http://musicbrainz.org/artist/09720eec-3871-49d5-932d-eb7542768cd3>
  schema:name "JIN" ;
  rdf:type schema:MusicGroup , schema:Person .

<http://musicbrainz.org/artist/ee27b2d8-648c-4a9d-a68c-e55066959975>
  schema:name "Megan Thee Stallion" ;
  rdf:type schema:MusicGroup , schema:Person .

<http://musicbrainz.org/release-group/0a2eb4c9-2bf1-496f-b74d-ff5684
```

```

schema:albumProductionType "http://schema.org/RemixAlbum" ;
schema:albumReleaseType "http://schema.org/SingleRelease" ;
schema:byArtist
  <http://musicbrainz.org/artist/0d79fe8e-ba27-4859-bb8c-2f255f346853> ,
  <http://musicbrainz.org/artist/ee27b2d8-648c-4a9d-a68c-e55066959975> ;
schema:creditedTo "BTS feat. Megan Thee Stallion" ;
schema:name "Butter" ;
rdf:type schema:MusicAlbum .

```

- (a) What (approximately) was the type that we put into the accept header? [1]
- (b) To indicate that someone is a member of a band in this model, the person is associated with a role using `schema:member` and then that role is associated with the group, also using `schema:member`. What is the full URL of the predicate `schema:member`? [1]
- (c) How many band members of BTS are listed in this snippet? [1]
- (d) Comment on the way the `schema:member` predicate is used in this context. [3]
- (e) What type(s) are associated with the entity with a `schema:name` of "JIN"? [1]
- (f) Consider the following SPARQL query

```

SELECT ?a ?b WHERE {
  mba:9fe8e-ba27-4859-bb8c-2f255f346853 schema:member ?c .
  ?c schema:startDate ?b ;
    schema:member ?d .
  ?d schema:name ?a .
}

```

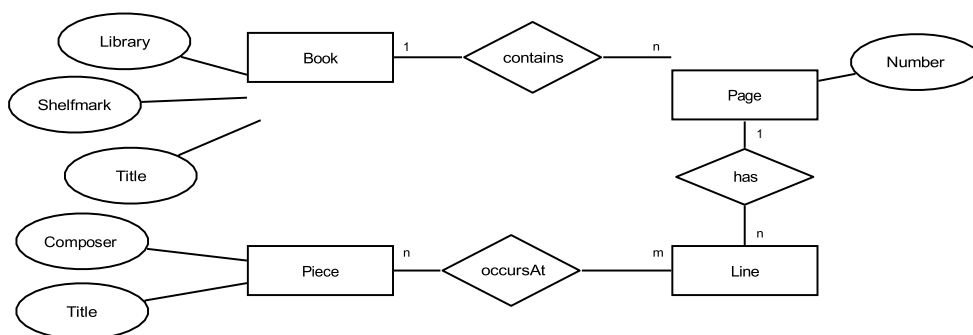
What prefixes need to be defined for this to work (give the full declarations)? [1]

- (g) What would the query return? [2]
- (h) This data represents an export from a relational database. Construct an ER diagram providing a model that could accommodate the instance data above. [6]
- (i) Give the CREATE TABLE commands for two tables based on your ER model. [4]
- (j) Suggest a MySQL query to check whether any band member in the database is recorded as joining before the founding date of their band. [5]

- (k) MusicBrainz make their data available as both a downloadable database dump and as Linked Data of the sort quoted above. What are the benefits and disadvantages of each approach? [5]

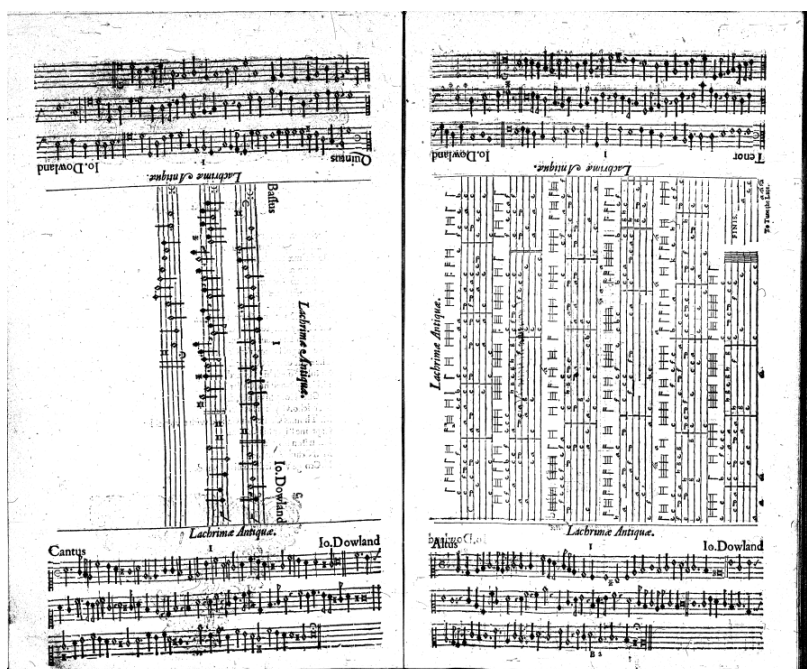
Question 4

A database of 16th-century European music records what pieces are located in which books, and where. A single book will contain many pages. A given musical piece can cover several lines of multiple pages. The Entity-Relationship diagram below shows an attempt to model this situation.



- (a) This model doesn't allow storing the order or co-ordinates for lines of music on a page, meaning that retrieving all the lines for a given piece will jumble them up. How could this be fixed?
- (b) Some books are published in tablebook format. These are designed to be placed on a flat table, with multiple performers standing around the book. An example is given below:

[3]



To add this to our model, we need to resolve two concerns. Firstly, there are multiple instrument or voice parts to a piece. Secondly, the page has regions, each with lines going in different directions. Add these aspects to the model.

[8]

- (c) List the tables, primary and foreign keys for a relational implementation of your modified model. [7]
- (d) Give a query to list pieces with the total number of lines of music that they occupy. [5]
- (e) Assess the suitability of this data structure for a relational model, and compare with ONE other database model from the course (XML-based tree database, document database or Linked data graph database). [7]

END OF PAPER