IMPERIAL COLLEGE LONDON

TIMED REMOTE ASSESSMENTS 2021-2022

BEng Honours Degree in Computing Part I
MEng Honours Degrees in Computing Part I
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant assessments for the Associateship of the City and Guilds of London Institute

PAPER COMP40007

INTRODUCTION TO DATABASES

Wednesday 11 May 2022, 10:00
Writing time: 80 minutes
Upload time: 25 minutes

Answer ALL TWO questions
Open book assessment

This time-limited remote assessment has been designed to be open book. You may use resources which have been identified by the examiner to complete the assessment and are included in the instructions for the examination. You must not use any additional resources when completing this assessment.

The use of the work of another student, past or present, constitutes plagiarism. Giving your work to another student to use constitutes an offence. Collusion is a form of plagiarism and will be treated in a similar manner. This is an individual assessment and thus should be completed solely by you. The College will investigate all instances where an examination or assessment offence is reported or suspected, using plagiarism software, vivas and other tools, and apply appropriate penalties to students. In all examinations we will analyse exam performance against previous performance and against data from previous years and use an evidence-based approach to maintain a fair and robust examination. As with all exams, the best strategy is to read the question carefully and answer as fully as possible, taking account of the time and number of marks available.

Paper contains 2 questions

Several parts of the following questions make use of the **mondial** relational database, a fragment of which is listed below. It contains information about countries, and the membership of countries in organisations.

For each organization, there is a record of the city and country in which that organisation is based, which if not present indicates that the organisation has no official base.

The is_member table records the relationship of countries to organisations, where types of membership include member for a full member, observer for non-members with a right to attend meetings, associate for partial members with some voting rights, *etc*.

The borders table records which countries share a land border, and the length of that border. Note that each pair of neighbouring countries appears only once in borders.

The percentage of the land area of each country that falls within a particular continent is recorded in encompasses.

organization

abbreviation	city?	country?	established?
AL	Cairo	ET	1945-03-22
C	London	GB	1931-12-31
CERN	Geneva	CH	1953-07-01
CSTO	Moscow	R	2002-10-07
EU	Brussels	B	1992-02-07
NATO	Brussels	B	1949-09-17
PCA	null	null	1899-07-29
WFTU	Prague	CZ	1945-10-03
⋮			

country

name	code	capital	area	population
Czech Republic	CZ	Prague	78,703	10,321,120
Switzerland	CH	Bern	41,290	7,207,060
Russia	R	Moscow	17,075,200	148,178,487
Belgium	B	Brussels	30,510	10,170,241
Turkey	TR	Ankara	780,580	62,484,478
United Kingdom	GB	London	244,820	58,489,975
Egypt	ET	Cairo	1,001,450	63,575,107
⋮				

encompasses

country	continent	percentage
CZ	Europe	100
CH	Europe	100
R	Europe	25
R	Asia	75
B	Europe	100
TR	Europe	3
TR	Asia	97
GB	Europe	100
ET	Asia	7
ET	Africa	93
⋮		

is_member

country	organization	type
CZ	CERN	member
CZ	EU	member
CZ	WFTU	member
CH	CERN	member
R	CERN	observer
B	CERN	member
B	EU	member
B	NATO	member
TR	CERN	observer
TR	NATO	member
TR	WFTU	member
GB	C	member
GB	CERN	member
GB	EU	member
GB	NATO	member
⋮		

borders

country1	country2	length
GR	TR	206
CZ	A	362
CZ	D	646
CZ	PL	658
FL	CH	41
SK	CZ	215
CH	F	573
CH	A	164
CH	D	334
CH	I	740
PL	R	206
UA	R	1,576
B	F	620
B	D	167
B	NL	450
L	B	148
TR	IR	499
⋮		

is_member(organization) $\overset{fk}{\Rightarrow}$ organization(abbreviation)
is_member(country) $\overset{fk}{\Rightarrow}$ country(code)
encompasses(country) $\overset{fk}{\Rightarrow}$ country(code)
borders(country1) $\overset{fk}{\Rightarrow}$ country(code)
borders(country2) $\overset{fk}{\Rightarrow}$ country(code)
organization(country) $\overset{fk}{\Rightarrow}$ country(code)

1 The following parts all refer to the **mondial** relational schema on Page 1.

a Write an RA query that returns the scheme (code) listing the code of all countries that have no land border with any other countries.

b Write an SQL query that returns the scheme (name) listing the name of countries that are not full members of any of the organisations that Turkey is a full member of.

c Consider the following SQL query:

```
SELECT name,
       capital
FROM   country
WHERE  population>1000000
       AND NOT EXISTS (SELECT *
                       FROM   organization
                       WHERE  country.code=organization.country
                       AND    country.capital=organization.city)
```

i) Give the result of the query for the subset of the **mondial** listed on Page 1 and explain the semantics of the query.

ii) Rewrite the SQL query into the RA

iii) Rewrite the SQL query into Datalog

d Consider the following SQL query:

```
SELECT name,
       MAX(length) AS longest
FROM   country
       JOIN borders
       ON   country.code IN (borders.country1,borders.country2)
GROUP BY name
```

Write an equivalent SQL query that makes no use of aggregate functions (such as MAX, MIN, *etc*, and makes no use of GROUP BY.

e Write an SQL query returning the scheme (name, continent, pc_area) listing the names of countries and the continents they are in, where pc_area is the land area of the country that falls within the continent compared to the total land area of all countries in that continent (expressed as a percentage). The listing should be restricted to contain countries which have at least 5% pc_area.

The five parts carry, respectively, 10%, 15%, 30%, 20%, and 25% of the marks.

2 a We wish to setup a new database containing information about bookings of rooms for courses at a university.

Each course is identified by a code, is given a name, and a department that is responsible for running the course. Some courses are also given a maximum capacity. Some courses have one member of staff called a leader appointed. For these leaders we need to record their id number, name, and email address. Leaders can oversee more than one course.

Rooms have a unique room number. For each room we know the seating capacity, and building in which it is situated.

Booking periods are identified by the combination of term, day of the week, and the hour of the day. We need to record for each booking period if it is to be only used for teaching.

Bookings are recorded for a course, in a given room, during a given booking period. For bookings we record the date they are made.

i) Design an ER$^{\mathcal{ADHKLMNOSVW}}$ schema to represent this new database.

ii) Map the ER schema you designed in (i) into a relational schema.

b Suppose that a relation $R(A, B, C, D, E, F, G, H)$ has the functional dependencies:
$S = \{AE \rightarrow AEF, B \rightarrow DC, D \rightarrow B, DH \rightarrow C, C \rightarrow H,$
$F \rightarrow E, G \rightarrow ACDH\}$.

Also suppose it has been proposed that the relation R be decomposed into $R_a(ADEFG), R_b(DBCH)$

i) Compute a minimum cover S_c of S.

ii) Identify and justify all the candidate keys of R.

iii) Is R_a and R_b a lossless decomposition of R and does it preserve the functional dependencies?

iv) Determine and justify if each of R_a and R_b are in 3NF. If the relations are not in 3NF, give an alternative decomposition of R that is in 3NF.

c The following histories describe the sequence of operations performed respectively by four transactions T_1–T_4.

$H_1 = r_1[c_{CZ}], w_1[c_{CZ}], r_1[c_R], w_1[c_R], c_1$
$H_2 = r_2[c_{CZ}], r_2[c_B], r_2[c_R], r_2[c_{GB}], c_2$
$H_3 = r_3[c_R], w_3[c_R], r_3[c_B], w_3[c_B], c_3$
$H_4 = r_4[c_{CZ}], r_4[c_B], r_4[c_R], r_4[c_{GB}], w_4[c_{GB}], c_4$

Answer the following questions with reference to these transactions:

i) Give a concurrent execution H_i of any two transactions that suffers an inconsistent analysis, but is also strict (ST) recoverable.

ii) Justify if there exists any pair of transactions for which it is impossible to write a concurrent execution H_n that is non-serialisable.

iii) Justify if there exists a pair of transactions for which there is a concurrent execution H_l that suffers a lost update, but where H_l is recoverable.

The three parts carry, respectively, 30%, 40%, and 30% of the marks.