VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wananga o te Upoko o te Ika a Maui

EXAMINATIONS — 2002
MID-YEAR

COMP 302

Database Systems

Time allowed: 3 Hours

Instructions: Answer all questions.
Make sure your answers are clear and to the point.
Calculators and foreign language dictionaries are allowed.
No reference material is allowed.
There are 180 possible marks on the exam.

CONTENTS:

Question 1. Enhanced Entity Relationship Data Model [25 marks]

Question 2. SQL and Relational Algebra [25 marks]

Question 3. Mapping ER to relational data model [30 marks]

Question 4. Functional Dependencies and Normalization [35 marks]

Question 5. Object-Oriented Databases [15 marks]

Question 6. Query Optimisation [38 marks]

Question 7. Concurrency Control [12 marks]

Attachment: Formulae for Computing Query Cost Estimate

Question 1. Enhanced Entity Relationship Data Model [25 marks]

Entertainment Ltd. decided to store information about musicians who perform on its albums as well as other company data in a database. The company has wisely chosen to hire you as a database designer. Draw an Enhanced Entity Relationship (EER) diagram that will correspond to the following description of a part of the real system.

- The company is made of departments. Each department is identified by *DepartmentId*, has a *Dep_Name*, and *Genre* (genre relates to the type of music albums that the department produces).
- Company *employees* and *musicians* are *persons*. Each person is identified by *PersonId*, has a *Per_Name*, and *Address.* Additionally, a musician has a *Type* (e.g. instrumentalist, vocalist), and a *Account_No*.
- Each *instrument* that is used in songs recorded at Entertainment Ltd is identified by a *Instr_Name* (e.g. guitar, flute,…) and has a *Musical_Key* (e.g. C, B-flat,…).
- Each album that is recorded on the Entertainment Inc. Ltd. label is identified by *AlbumId*, has a *Alb_Title*, a *Copy_Right* and a *Format* (e.g. CD or MC).
- Each song recorded at Entertainment Inc. Ltd. appears on exactly one album and is identified by *AlbumId* and *SongOrdNo*. Additionally a song has a *Song_Title*.
- Each employee works for exactly one department.
- One department may employ many employees, but there may exist a department without any employees.
- A department may engage several musicians and a musician is engaged by at least one department. The property of department-musician association is *Period*.
- Each musician may play several instruments, but there may exist musicians who do not play any instruments.
- More than one musician may play any given instrument, but there may exist an instrument that is not played by any musician.
- Each album contains at least one song, and a song appears on exactly one album.
- Each song is performed by at least one musician, and each musician may perform many songs. However, there may exist a musician who does not perform any songs.
- Each album has exactly one musician who acts as its producer. A musician may produce several albums, but not all musicians act as producers.

IMPORTANT NOTE:

When drawing the requested EER diagram use the notation and conventions that we introduced in lectures. If for any reason you prefer to use a different notation, provide clear instructions for the mapping to the notation introduced in lectures.

Question 2. SQL and Relational Algebra [25 marks]

Consider the following set of relation schemas for a database named *BOAT_HIRE* that keeps track of sailors, boats, and reservations of boats made by sailors.

Sailor ({*SailId*, *Sail_Name*, *Rating*, *Age* }, {*SailId* }),

Boat ({*BoatId*, *Boat_Name*, *Color* }, {*BoatId* }),

Reserves ({*SailId*, *BoatId*, *Date* }, {*SailId* + *BoatId* + *Date*}),

a) Write the following SQL queries on the *BOAT_HIRE* database:

   i. Retrieve sailor id's and sailor names of sailors who have not reserved a boat. [5 marks]

   ii. Retrieve sailor id's and sailor names of sailors that are older than 18 and have reserved a red boat for to_day (suppose the DBMS supports a function CURRENT_DATE that returns current date). [5 marks]

   iii. Find the sailor id's and sailor names of sailors who are older than the oldest sailor with a rating of 10. [10 marks]

b) Write the following relational algebra queries on the *BOAT_HIRE* database:

   i. Retrieve sailor id's and sailor names of sailors who are older than 18 and have reserved a red boat. [5 marks]

Question 3. Mapping ER to relational data model [30 marks]

Consider the EER data model that describes part of a conceptual schema for a university database. The diagram is shown in the Figure 2.1. Note that the entity type attributes are given separately in the Figure 2.2, in the form

$$ENTITY_TYPE_NAME\ (Atribute_1, \ldots, Attribute_n),$$

and that entity type keys are underlined, whereas weak entity type partial keys are underlined with dashes.

The short description of the part of the real world represented by the EER diagram is:

General staff, academic staff, and students are persons. General staff and academic staff work for a department. A department owns some courses. An academic staff lectures courses. Students take courses. Courses use text books that are written by their authors.

Map the university database conceptual EER schema into a relational database schema. When answering this question define:

a) A set of relation schemas together with their keys, [15 marks]

b) Null value constraints of the form *Null* (*Relation_Schema_Name*, *Foreign_Key*) for each relation schema foreign key pair. Let the constraint value be *N* if the foreign key is not allowed to have a null value, and *Y* otherwise. [5 marks]

c) A set of non redundant referential integrity constraints, and [5 marks]

d) The highest normal form your relational database schema is in. [5 marks]

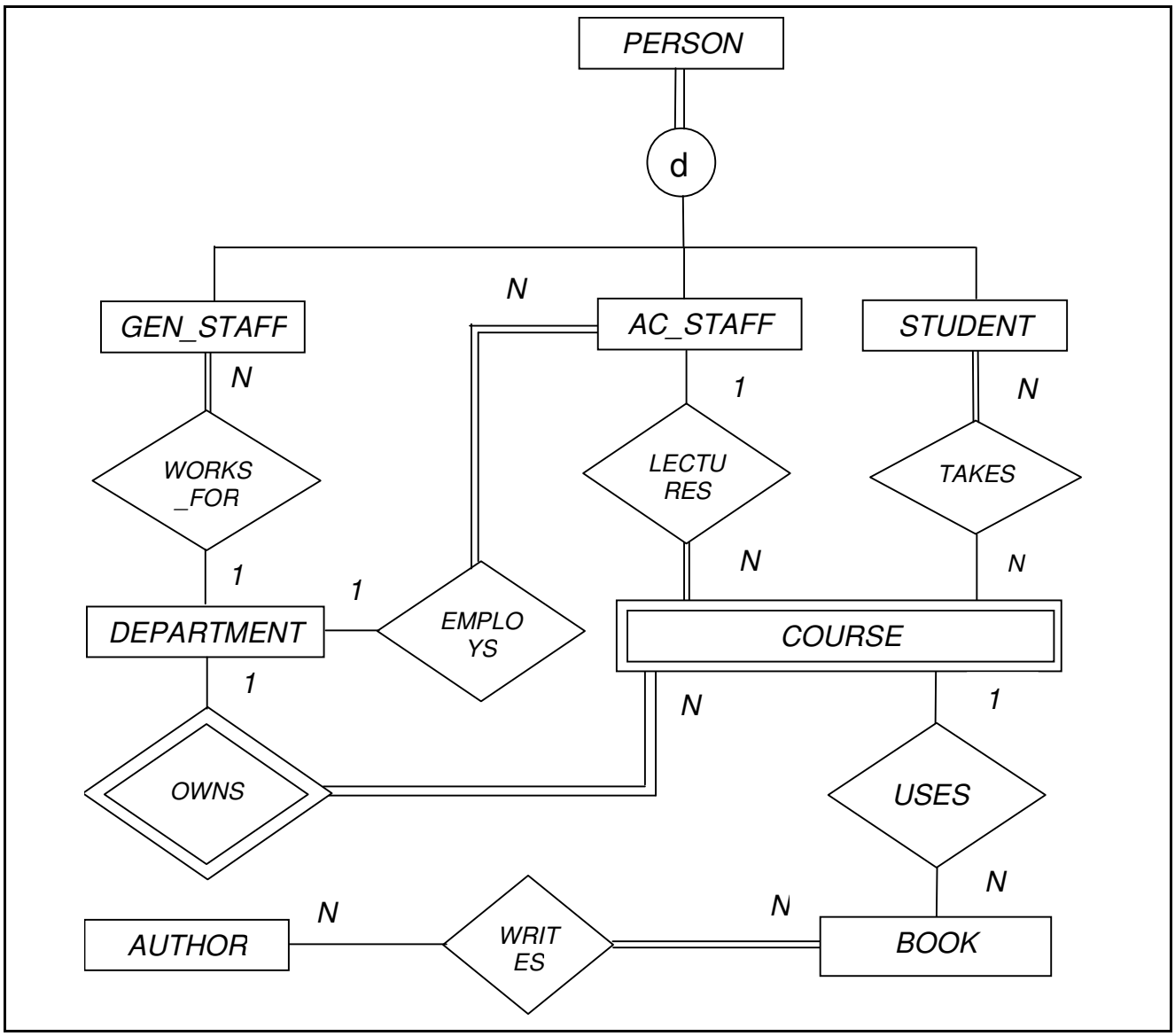

Figure 2.1

PERSON(<u>*PerId*</u>, *Per_Name*, *Per_Address*)
GEN_STAFF(*Gen_Pos*)
AC_STAFF(*Position*, *Ac_Qual*)
DEPARTMENT(<u>*DeptId*</u>, *Dept_Name*)
STUDENT(*Major*, *NoOfPoints*)
COURSE(<u>*Cour_No*</u>, *Cour_Name*)
BOOK(<u>*ISBN*</u>, *Title*, *Publisher*)
WRITES(*AuthSeqNo*)
AUTHOR(<u>*AuthId*</u>, *Auth_Name*)

Figure 2.2.

COMP 302 **continued**

Question 4. Functional Dependencies and Normalization [35 marks]

a) Express the following real world constraints using functional dependencies:

i. Each course (*Course*) belongs to one department (*Department*). [1 mark]

ii. Each staff (*Staff*) member works for one department (*Department*). [1 mark]

iii. In a term (*Term*) of a year (*Year*) a staff member (*Staff*) coordinates at most one course (*Course*). [3 marks]

b) Let a set of functional dependencies (fd's)

$$F_1 = \{A \rightarrow BC,\ C \rightarrow DEG\}$$

be given. Apply the decomposition inference rule on the set F_1 to produce a new set of fd's F_1', where each fd will have only one attribute on its right hand side. [2 marks]

c) Let a set of functional dependencies

$$F_2 = \{AB \rightarrow C,\ DEG \rightarrow H,\ A \rightarrow B,\ DG \rightarrow H\}$$

be given. Transform the set F_2 into a new set of fd's F_2', where each fd will be left reduced (ie will have no redundant attribute on its left hand side). [5 marks]

d) Let a set of left reduced functional dependencies

$$F_3 = \{AB \rightarrow C,\ C \rightarrow D,\ C \rightarrow C,\ AB \rightarrow D,\ CE \rightarrow G,\ DE \rightarrow G\}$$

be given. Transform the set F_3 into a new, non redundant set of fd's F_3'. [8 marks]

e) Let a set of left reduced and non redundant functional dependencies

$$F_4 = \{A \rightarrow B,\ A \rightarrow C,\ A \rightarrow D,\ B \rightarrow E,\ B \rightarrow G,\ GH \rightarrow I\}$$

be given. Partition the set F_4 into subsets P_i, $i = 1, 2, \ldots$, where each subset will contain only those functional dependencies from F_4 that contain the same left hand side. [2 marks]

Build a set S of relation schemas by transforming each subset P_i into a relation schema of the form $N_i(R, K)$, where N_i is the relation schema name, R is the set of attributes, and K is the set of keys. [2 marks]

What is the highest normal form of the relation schema set S? [3 marks]

f) Let a universal relation schema (U, F) be given, where

$$U = \{A, B, C, D, E, G\} \text{ and } F = \{AG \rightarrow C,\ C \rightarrow D,\ E \rightarrow D\}.$$

Suppose that starting from (U, F) the following set

$$S = \{N_1(\{A, G, C\}, \{AG\}),\ N_2(\{C, D\}, \{C\}),\ N_3(\{D, E\}, \{E\})\}$$

of BCNF relation schemas has been produced.

If you consider S is a lossless join decomposition of (U, F) explain why it is. If you consider S is not a lossless join decomposition of (U, F), explain why it is not, and transform it into a new set of relation schemas S' that will be a BCNF and lossless join decomposition of (U, F). [7 marks]

g) Questions b) to f) represent the steps of a normalization algorithm. What is the name of that algorithm? [1 mark]

Question 5. Object-Oriented Databases [15 marks]

Relational and object - oriented approaches differ in many aspects. Comment and briefly explain how they differ with regard to:

a) Object and tuple identification and association, [3 marks]

b) Data types, [3 marks]

c) Object and tuple structure, [3 marks]

d) Mechanisms to achieve persistence, and [3 marks]

e) Type of data definition and data manipulation languages in current use. [3 marks]

Question 6. Query Optimisation [38 marks]

Consider the description of the *BOAT_HIRE* database schema, given in the Table 5.1, and the corresponding parameters of the physical database structure, given in the Table 5.2. In the Table 5.2, *L* is the tuple length (size), *f* is the blocking factor, *B* is the block size, *r* is the number of tuples, and *x* is either the primary, or secondary key index height. The values of all the Table 5.2 parameters pertain to the base relations.

Attribute	Data Type	Max. Length	Null	Default
Relation name: *Sailor*, Primary Key: *SailorId*				
SailId	Int	4	N	
Sail_Name	Char	40	N	
Rating	Int	2	N	
Age	Int	2	N	
Relation name: *Boat*, Primary Key: *BoatId*				
BoatId	Int	4	N	
Boat_Name	Char	15	Y	
Color	Char	10	N	
Relation name: *Reserves*, Primary Key: *SailId* + *BoatId* + *Date*				
SailId	Int	4	N	
BoatId	Int	4	N	
Day	Date	8	N	

Table 5.1.

Relation schema	*L*	*f*	*B*	*r*	*x* (primary)	*x* (secondary)
Sailor	*48*	*10*	*480*	*1300*	*2*	*1*
Boat	*29*	*17*	*493*	*100*	*1*	-
Reserves	*16*	*31*	*496*	*26000*	*3*	-

Table 5.2.

a) Draw the heuristic optimization tree that corresponds to the SQL query:

```
SELECT s.SailId,Sail_Name, b.BoatId
FROM (Sailor s NATURAL JOIN Reserves r) NATURAL JOIN Boat b
WHERE Color = “red” AND Day = CURRENT_DATE;
```

[8 marks]

b) Suppose:

- Each sailor may have one of 10 rating values, and there are an approximately equal number of sailors with each rating value.
- Only first names of sailors are recorded, and there are *100* different first names among all sailors,
- The intermediate results of the query evaluation are materialized,
- The final result of the query is materialized,
- The size of each intermediate result block should not exceed 500 bytes,
- There is a buffer pool of 6000 bytes provided for query processing in the main memory,
- There are primary indexes on *Sailor.SailId*, *Boat.BoatId*, and *Reserves*.(*SailId*, *BoatId*, *Day*) available, and
- There is a secondary index on *Sailor.Rating* available.

i. Calculate the lowest execution cost of the query

SELECT * FROM Sailor WHERE Rating = 5; [5 marks]

ii. Calculate the lowest execution cost of the query

SELECT DISTINCT Sail_Name FROM Sailor; [9 marks]

iii. Calculate the lowest execution cost of the query

SELECT * FROM Sailor s, Reserves r

WHERE s.SailId = r.SailId; [16 marks]

Question 7. Concurrency Control [12 marks]

Consider the diagram in Figure 7.1 showing the concurrent execution of two transactions.

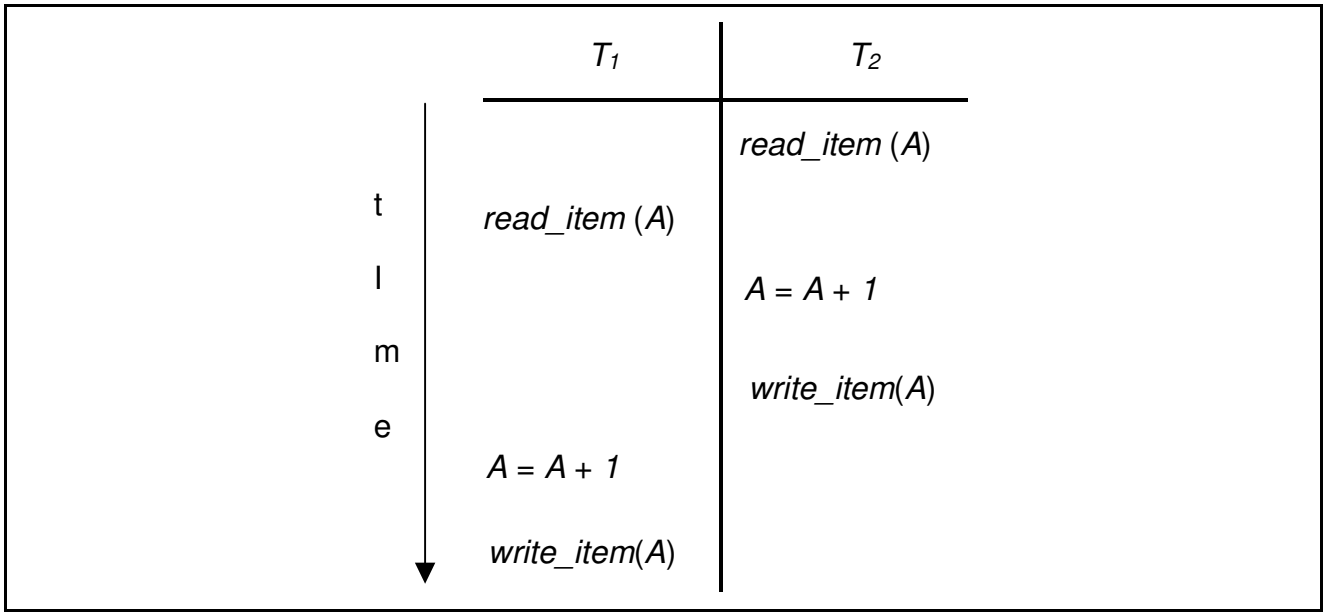

Figure 7.1

a) Concurrent execution of these two transactions leads to a well-known concurrency anomaly. What is the name of this anomaly? [1 mark]

b) Suppose there are no locks applied, and suppose the database item *A* had the value *0* before the start of transactions T_1 and T_2. After a concurrent execution of these two transactions the database will be left in an inconsistent state.

   i. What will be the contents of item *A*? [1 mark]

   ii. What should be the contents of item *A*.? [1 mark]

c) Show how conservative two - phase locking protocol ensures conflict serializability of T_1 and T_2, and justify that claim using a precedence graph. [9 marks]

* * * * * * * * * * * * * * * * * * * *

COMP 302 **end**

FORMULAE FOR COMPUTING QUERY COST ESTIMATE

Blocking factor: $f = \lfloor B / L \rfloor$

Number of blocks: $b = \lceil r / f \rceil$

Selection cardinality of the attribute A : $s(A) = r / d(A)$, where $d(A)$ is the number of different A values

C (project) = $b_1 + b_2$

C(project_distinct) = $b_1 + b_2 + 2b_2(1 + \lceil \log_m b_2 \rceil) + b_2 + b_3$, $m = n - 1$

C (select_linear) = $b_1 + \lceil s / f \rceil$

C (select_sec_key) = $x + s + \lceil s / f \rceil$

C (nested_join) = $b_1 + b_2 \lceil b_1 / (n - 2) \rceil + \lceil js^* r_1^* r_2 / f \rceil$

C (single_join) = $b_1 + r_1^* f(index_2) + \lceil js^* r_1^* r_2 / f \rceil$ (4 buffers minimally required)

C (sort_join) = $b_1(3 + 2\lceil \log_m b_1 \rceil) + b_2(3 + 2\lceil \log_m b_2 \rceil) + \lceil js^* r_1^* r_2 / f \rceil$, $m = n - 1$

C(partition_join) = $3(b_1 + b_2) + \lceil js^* r_1^* r_2 / f \rceil$, $n \geq \lceil (3 + (1 + 4b_1)^{1/2})/2 \rceil$, $b_1 < b_2$

C (sort) = $2b(1 + \lceil \log_m b \rceil)$

f (*index*):

primary index: $f(index) = x + 1$

secondary index: $f(index)$: $x + s$